

Instituto Tecnológico de Costa Rica

Ingeniería en Computación

Campus San Carlos

Segundo Proyecto Programado

Programación Orientada a Objetos

Profesor: Oscar Víquez Acuña.

Sistema: “Gimnasio”

Estudiantes:

Justin Alonso Martínez Molina 201902704

Roosevelt Alejandro Pérez González 2022119527

2023

Análisis del Problema:

El problema planteado consiste en el diseño y desarrollo de un sistema de información de rutinas de entrenamiento personales. El sistema permitirá a los usuarios crear, ejecutar y gestionar rutinas personalizadas, realizar un seguimiento de su progreso y recibir notificaciones de mejora. Para resolver el problema, se deben considerar requisitos técnicos, como la ocultación de detalles de implementación, el manejo de excepciones, la creación de un diagrama de clases y la implementación de patrones de diseño adecuados.

El problema específico que se debe abordar es el diseño y desarrollo de un sistema que permita a los usuarios crear y gestionar rutinas de entrenamiento de gimnasio, así como realizar un seguimiento diario de su ejecución y recibir alertas cuando superen sus propias marcas. Este sistema se enfocará en proporcionar una interfaz de usuario intuitiva y fácil de utilizar.

El sistema deberá permitir al usuario crear rutinas personalizadas, que consisten en conjuntos de ejercicios. Cada rutina estará identificada por un nombre único y contendrá una cantidad variable de ejercicios seleccionados de una lista de ejercicios disponibles. Para cada sección, se deberán proporcionar al menos cuatro ejercicios diferentes, seleccionados por el programador. Cada ejercicio contendrá una descripción, una imagen ilustrativa, y una cantidad de repeticiones sugerida.

Además de las rutinas personalizadas, el sistema deberá incluir algunas rutinas predefinidas. Los usuarios tendrán la opción de crear nuevas rutinas, especificando el tipo y la cantidad de ejercicios deseados.

Una vez que las rutinas estén disponibles, el sistema permitirá a los usuarios ejecutar instancias de estas, en diferentes momentos. Por ejemplo, un usuario podría seleccionar la rutina denominada "Quiebra piernas" para un día específico y registrar los datos de cantidad de repeticiones para cada ejercicio de esa rutina. Si el usuario logra mejorar su rendimiento en uno o más ejercicios, recibirá una notificación felicitándolo por su mejora.

El sistema también permitirá a los usuarios guardar y consultar las rutinas realizadas, independientemente de si se han realizado una o varias veces. Para este propósito, se deberá almacenar la fecha y hora en que se completó cada rutina.

En términos técnicos, se requerirá la implementación de un sistema que cumpla con ciertos requisitos y restricciones. Por ejemplo, se deberá ocultar los detalles de implementación innecesarios utilizando los modificadores de acceso adecuados. Además, se deberá realizar un diagrama de clases para planificar el diseño del sistema antes de la implementación. La gestión de excepciones será crucial para garantizar la ejecución correcta y controlada del sistema, ya sea utilizando excepciones existentes o creando excepciones personalizadas. Asimismo, se deberá aplicar al menos un patrón de diseño, distinto al patrón Singleton, para mejorar la estructura y mantenibilidad del código.

Objetivo General:

1. Desarrollar un sistema de información de rutinas de entrenamiento personales en Java que permita la creación, ejecución y seguimiento diario de las rutinas, así como el registro de mejoras y la posibilidad de guardar y consultar las rutinas realizadas por un único usuario.

Objetivos Específicos:

1. Implementar un sistema que le permita al usuario lograr crear una rutina y ejercicios a su gusto.
2. Lograr permitirle al usuario realizar estas rutinas, ya sean rutinas o ejercicios que le brindará la aplicación o creados por el usuario.
3. Incorporar un sistema de seguimiento y registro de mejoras, que notifique al usuario cuando haya superado sus propias marcas.

Análisis de la Solución planteada:

En la solución planteada para el proyecto de Programación Orientada a Objetos, se logró cumplir con la mayoría de los requisitos establecidos por el profesor. Se implementó adecuadamente el encapsulamiento de los métodos y clases pertinentes, lo que contribuye a la coherencia y seguridad de los datos. Además, se creó un diagrama de clases que permite visualizar de manera clara la estructura del sistema y las relaciones entre las clases. Asimismo, se incluyó un control adecuado del manejo de excepciones, lo que asegura una respuesta adecuada ante posibles errores en tiempo de ejecución. Por otro lado, se pudo generar un archivo ejecutable .jar, facilitando la distribución y ejecución de la aplicación en diferentes entornos. A pesar de esto, no se pudo implementar un patrón de diseño en la solución, lo cual podría haber mejorado aún más la estructura y reutilización del código. Asimismo, en los ejercicios no se incluyeron imágenes, lo cual podría haber enriquecido la experiencia del usuario. Finalmente, no se informa al usuario cuando se establece un nuevo récord, lo cual podría haber proporcionado una retroalimentación adicional y mayor satisfacción al usuario. Para proyectos futuros, es recomendable considerar la aplicación de patrones de diseño apropiados, la inclusión de imágenes cuando sea necesario y la implementación de notificaciones para informar sobre nuevos récords. Además, es importante destacar que el código desarrollado en esta solución ha sido diseñado de manera modular y estructurada, lo que lo hace apto para recibir futuras mejoras y expansiones. Aunque en el tiempo disponible no fue posible implementar todos los requisitos solicitados, es factible considerar su incorporación en futuras iteraciones del proyecto. Con un tiempo adicional y una planificación adecuada, se podrían abordar los aspectos pendientes, como la implementación de patrones de diseño, la inclusión de imágenes en los ejercicios y la notificación de nuevos récords. Este enfoque permitiría una evolución continua de la solución, garantizando su adaptabilidad y satisfaciendo de manera completa los requisitos establecidos.

Archivos		
m	Archivos()	
m	leerUsuario(String, SistemaGimnasio)	void
m	leerArchivo(String)	void
m	escribirRegistrosBinario(String, RegistroRutinas)	void
m	anexarAArchivo(String, String)	void
m	anexarAEjercicio(String, Ejercicio, SistemaGimnasio)	void
m	lecturaRutinaBinario(String, SistemaGimnasio)	void
m	escribirRutinasBinarios(String, Rutina)	void
m	escribirEjerciciosBinarios(String, Ejercicio)	void
m	crearArchivo(String)	void
m	lecturaEjerciciosBinario(String, SistemaGimnasio)	void
m	escribirArchivo(String, String)	void
m	leerEjercicios(String, SistemaGimnasio)	void
m	lecturaRegistroBinario(String, SistemaGimnasio)	void

Ejercicio		
m	Ejercicio(String, String, String, String)	
m	Ejercicio(String, String, String, String, String)	
f	imagen	BufferedImage
f	descripcion	String
f	seccionDelCuerpo	String
f	repeticiones	int
f	nombre	String
m	toString()	String
p	descripcion	String
p	repeticiones	int
p	seccionDelCuerpo	String
p	imagen	BufferedImage
p	nombre	String

RegistroRutinas		
m	RegistroRutinas(Rutina, Date)	
f	tiempoTotal	long
f	informacion	String
f	rutina	Rutina
f	fechalNicio	Date
m	toString()	String
m	finalizarRutina()	void
m	iniciarRutina()	void
p	rutina	Rutina
p	informacion	String
p	fechalNicio	Date
p	tiempoTotal	long

SistemaGimnasio		
m	SistemaGimnasio()	
f	usuario	Usuario
f	ejerciciosDisponibles	List<Ejercicio>
f	registrosRutinas	List
f	rutinasDisponibles	List<Rutina>
m	crearUsuario(String, String, String, String)	void
m	actualizarRutina()	void
p	ejerciciosDisponibles	List<Ejercicio>
p	registrosRutinas	List<RegistroRutinas>
p	usuario	Usuario
p	rutinasDisponibles	List<Rutina>

Usuario		
m	Usuario(String, String, String, String)	
f	edad	int
f	peso	double
f	altura	double
f	nombre	String
p	peso	double
p	edad	int
p	altura	double
p	nombre	String

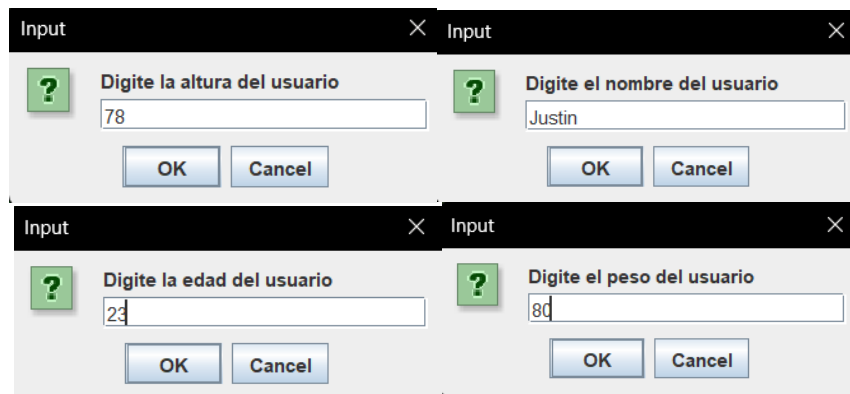
Rutina		
m	Rutina(String)	
f	nombre	String
f	listaDeEjercicio	List<Ejercicio>
m	eliminarEjercicio(String)	void
m	toString()	String
m	agregarEjercicio(Ejercicio)	void
p	nombre	String
p	listaDeEjercicio	List<Ejercicio>

Main		
m	Main()	
m	main(String[])	void

Resultados Obtenidos:

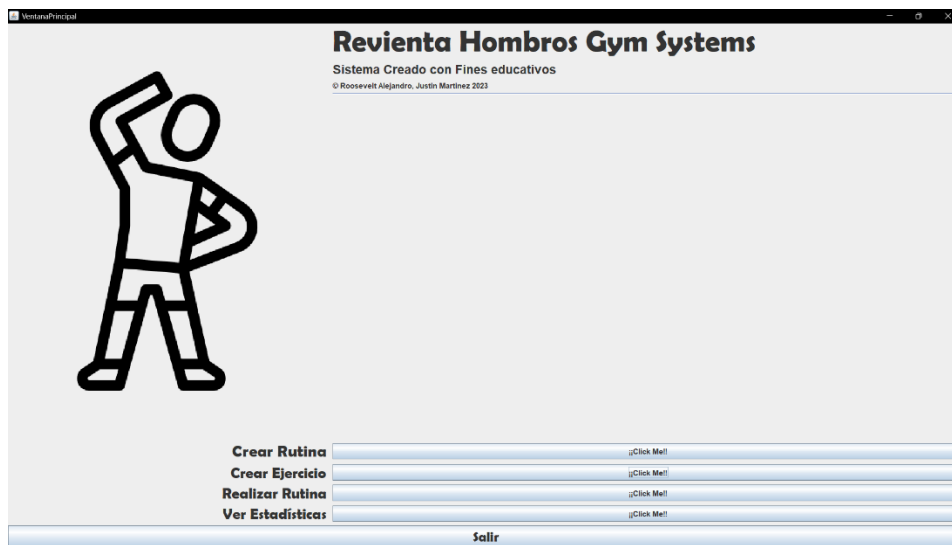
Cuando se ingresa por primera vez al sistema, este recibe al usuario mediante un login, en el cual se le realizan las siguientes preguntas:

- Nombre de usuario.
- Edad de usuario.
- Peso de usuario.
- Altura de usuario.



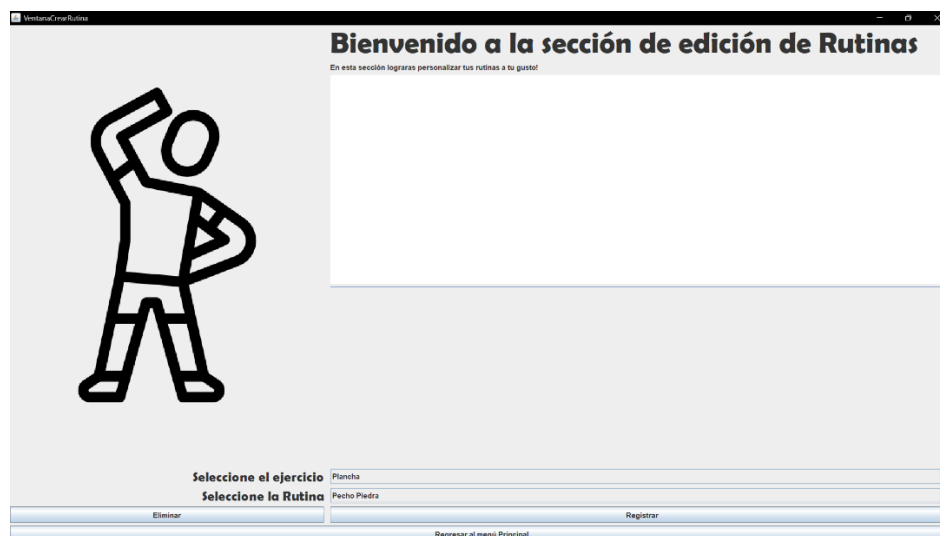
The image shows four separate 'Input' dialog boxes arranged in a 2x2 grid. Each dialog has a title bar with a close button (X), a green question mark icon, a text prompt, an input field, and 'OK' and 'Cancel' buttons. The prompts and values are: Top-left: 'Digite la altura del usuario' with value '78'; Top-right: 'Digite el nombre del usuario' with value 'Justin'; Bottom-left: 'Digite la edad del usuario' with value '23'; Bottom-right: 'Digite el peso del usuario' with value '80'.

Seguidamente se implementó un menú principal en el que se presentan las opciones generales del programa:



The main menu window has a title bar 'VentanaPrincipal'. The header area contains the title 'Revienta Hombros Gym Systems', the subtitle 'Sistema Creado con Fines educativos', and a copyright notice '© Roosevelt Alejandro, Justin Martinez 2023'. On the left is a stick figure icon of a person performing a shoulder press. On the right is a large empty rectangular area. At the bottom, there is a menu with five items: 'Crear Rutina', 'Crear Ejercicio', 'Realizar Rutina', 'Ver Estadísticas', and 'Salir'. Each item has a corresponding '¡Click Me!!' button to its right.

Esta pantalla de inicio cuenta con 5 opciones, la primer opción del menú “Crear Rutina” dirige al usuario a una nueva pantalla, esta pantalla está hecha para que el usuario pueda crear su rutina, y pueda también, ir observando los ejercicios que va a realizar en esta rutina de ejercicio, también, se puede agregar ejercicios a las rutinas, en el manual de usuario se dará una mejor explicación.



The routine editing window has a title bar 'VentanaCrearRutina'. The header area contains the title 'Bienvenido a la sección de edición de Rutinas' and a subtitle 'En esta sección lograras personalizar tus rutinas a tu gusto!'. On the left is the same stick figure icon. On the right is a large empty rectangular area. At the bottom, there are two dropdown menus: 'Seleccione el ejercicio' (with 'Plancha' selected) and 'Seleccione la Rutina' (with 'Pecho Piedra' selected). Below these are three buttons: 'Eliminar', 'Registrar', and 'Regresar al menú Principal'.

Para lograr salir de esta ventana se colocó un botón, este le permite al usuario salir de esta ventana y volver al menú principal.

Seguidamente en el menú, se encuentra la opción de “Crear Ejercicio”, al darle click, este botón redirige al usuario a una nueva ventana, la cual le permite crear ejercicios a su conveniencia.



Estos que se creen nuevos, se almacenarán en la lista de ejercicios para luego en la ventana de crear rutina, aparezcan en la lista de ejercicios que se deseen agregar a esta rutina, para ello luego de describirlo, debe de darle al botón de agregar ejercicio. Esto le permite al usuario tener su propia rutina de ejercicios. Cuando se desee regresar al menú principal, solamente el usuario debe de darle click al botón de regresar al menú principal. En el menú principal, hay un tercer botón que dice, “Realizar Rutina” esta ventana que se abre al presionar este, lo que realiza es que el usuario pueda hacer uso de sus ejercicios y rutinas implementadas.



Primero, se debe de elegir una rutina, luego, se presenta en un cuadro en la parte superior de la ventana, seguido a eso, el usuario debe de darle click al botón de “Realizar Rutina”, para así empezar a hacer sus debidos ejercicios. El sistema lleva un tiempo luego de haber realizado esta acción (darle al botón de realizar rutina), con esto el usuario puede observar el tiempo en el que hizo esta rutina. Para lograr terminar con esto, debe de darle al botón de “Detener Rutina” esta se guardará y se mostrará en la ventana de “Estadísticas”, la que se encuentra en el menú principal.



Esta ventana de estadísticas permite que el usuario pueda observar los siguientes datos durante su actividad: - Nombre de rutina - Día – Mes - Tiempo de ejecución – Año

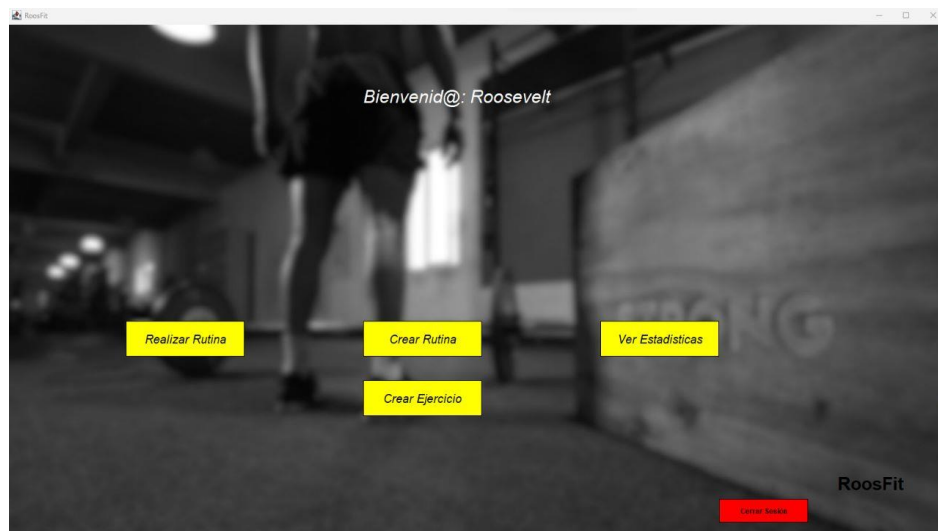
Por último, se encuentra el botón de “Salir” que le permite al usuario salir del sistema y cerrarlo. Cabe destacar que, si se desea volver a ingresar al sistema, os se mantienen guardados, así como el login, es decir, no hace falta volver a hacer el login al sistema.

Clases.	Completado S/N.	Porcentaje de acabado.
Crear Rutina	S	
Crear Ejercicio	S	100%
Realizar Ejercicio	S	100%
Borrar ejercicios	S	100%
Ver Estadísticas	S	50%
Salir	S	100%
Patrón de Diseño.	N	0%
Puntos adicionales	N	0%

Conclusiones y recomendaciones:

Es importante destacar los siguientes puntos que se pudieron haber implementado, pero por alguna circunstancia no se logró con el acabado.

- El sistema pueda documentar imágenes: se nos complicó el implementar imágenes, ya qué en un sistema anterior a este, logramos realizarlo:



- La interfaz puede ser más amigable con el usuario: Uno de los mayores problemas que se tuvo al iniciar el proyecto, fue la falta de conocimiento en el tema, pues se empezó por hacer la parte de implementación del código primero que la interfaz gráfica. Causando que nos quedáramos sin tiempo para dedicarle más a esta parte del trabajo.
- Implementar las funcionalidades que hicieron falta: Por ejemplo algunas de las funcionalidades que faltaron fueron, romper récord, “felicitaciones” cuando se lograba romper el récord, que se haya implementado algún patrón de diseño como el que se tenía pensado, en este caso se quería implementar el patrón de diseño “observer”. De igual forma, no se logró completar por falta de tiempo, es decir, el equipo de trabajo dedicó más tiempo a lograr entender y comprender como funcionaba el proyecto e implementar las soluciones a este. Quedándose sin tiempo para completar estas cosas que hacen falta.
- Eliminar rutinas, agregar rutinas por nombre.

Estas funciones que quedaron sin resolver quedan en la posibilidad de lograr su implementación al sistema, es decir, este cuenta con la disposición de que se le pueda añadir fácilmente si se desea continuar, esto podría ayudar más que todo con la enseñanza y la práctica de los estudiantes a cargo del sistema.

También, cabe recalcar que el sistema cuenta con más de la mitad de lo solicitado, y cumple con más de la mitad, por ende, se considera un sistema funcional, que con mayor organización, conocimiento y experiencia, este trabajo se le pudo realizar un mejor acabado. También es importante destacar que en la imagen anterior donde se muestra una pantalla de inicio con una imagen y colores si había logrado implementar el patrón de diseño y se había logrado realizar un acabado visual más estético

Link del manual de usuario: <https://youtu.be/9HPVPp-wqSo>