

PROYECTO: SEGUIMIENTO DE ACTIVIDADES FÍSICAS

CS1112 - PROGRAMACIÓN II



LABORATORIO 2.09

October 11, 2023

1 ENUNCIADO DEL PROYECTO

El objetivo de este proyecto es desarrollar un aplicativo de consola en el lenguaje de programación C++ que permita medir el avance físico de varias personas a través de la introducción y seguimiento de sus datos relacionados con el estado físico y las actividades de ejercicio. El aplicativo deberá ser capaz de calcular diversas métricas de fitness y generar reportes detallados con estadísticas relevantes para evaluar el progreso de los usuarios.

1.1 REQUERIMIENTOS DEL PROYECTO

Su programa debe implementar un menú que permita seleccionar las siguientes funcionalidades tomando las siguientes consideraciones:

1. Registro de Usuarios:

- **Datos Personales:** Los usuarios podrán ingresar su nombre, edad, género, altura, peso y cualquier otra información relevante para el cálculo del fitness.
- **Validación de Datos:** El aplicativo debe validar la entrada del usuario para asegurarse de que los datos ingresados sean coherentes y válidos.

2. Seguimiento de Actividades:

- **Tipo de Ejercicio:** Los usuarios podrán especificar el tipo de actividad física que realizaron, como correr, nadar, levantar pesas, etc.
- **Duración e Intensidad:** Deberán ingresar la duración de la actividad y la intensidad (por ejemplo, velocidad en caso de correr, o peso en caso de levantar pesas).
- **Frecuencia:** Podrán registrar la frecuencia con la que realizan estas actividades (por día, por semana, etc.).

3. Cálculo del Avance Fit:

- **Índice de Masa Corporal (IMC):** Se calculará utilizando la fórmula $IMC = \frac{peso(kg)}{altura(m)^2}$. Esto proporcionará una indicación básica del peso corporal en relación con la altura.
- **Frecuencia Cardíaca Promedio:** Si se proporciona información sobre la frecuencia cardíaca durante el ejercicio, se calculará la frecuencia cardíaca promedio.
- **Otras Métricas:** Dependiendo de la disponibilidad de datos, se podrían calcular otras métricas como la distancia recorrida, las repeticiones de ejercicio, etc(Opcional)

4. Generación de Reportes Estadísticos:

- **Gráficos de Progreso:** Se generarán gráficos que muestren la evolución del IMC, las calorías quemadas y otras métricas a lo largo del tiempo para visualizar el progreso.
- **Historial Detallado:** Se proporcionará un historial detallado de todas las actividades ingresadas y los cálculos realizados para que los usuarios puedan revisar su progreso en cualquier momento.

- Debe haber la opción de poder generar un reporte por usuario o si se desea de todos los usuarios.
- El aplicativo debe permitir exportar los distintos reportes en un documento como por ejemplo csv, docx o txt. Cualquiera de los 3 tipos de documentos estaría bien, solo basta implementar uno de ellos.

5. Funcionalidades Adicionales:

- **Establecimiento de Metas:** Los usuarios podrán establecer metas específicas, como perder peso, ganar masa muscular, o mejorar la resistencia, y el aplicativo mostrará el progreso hacia esas metas.
- **Consejos de Salud y Nutrición:** El aplicativo podría proporcionar consejos sobre hábitos saludables, nutrición balanceada y formas de mantenerse motivado para alcanzar los objetivos de fitness.

Su programa debe implementar **Validaciones** según se indica a continuación:

- Se debe validar que solamente se deba ingresar opciones que estén dentro del rango del menú.
- El aplicativo debe validar los valores que se están solicitando para asegurarse de que los datos ingresados sean coherentes y válidos.

2 PARTES DEL PROYECTO

2.1 PROGRAMACIÓN/REQUISITOS TÉCNICOS

Debe desarrollar el programa en lenguaje C++ en modo consola, como se ha practicado en clase. Debe utilizar varios módulos para distribuir su código en clases y dividir el programa en funciones.

Para las clases, funciones y variables debe utilizar nombres que describan el propósito de las mismas. Puede utilizar librerías para generar los valores aleatorios en caso se requiera.

2.2 DEMOSTRACIÓN

- El **proyecto se entrega el día 19 de Noviembre antes de las 20 hrs. via gradescope.** La entrega es grupal, es decir solo sube un alumno del grupo.
- **Las exposiciones se realizarán de manera grupal en la sesión 15B (Jueves 23 de Noviembre).**
- Cada grupo dispone de **5 minutos para que puedan exponer las funcionalidades,** luego hay 5 minutos de preguntas sobre el proyecto.

2.3 ENTREGA DEL PROYECTO

- Durante el ciclo, se realizará un solo un proyecto que corresponde al rubro de evaluación P1.

- El proyecto se realizará de manera grupal, los grupos serán de 5 alumnos y su profesor determina el modo en que se formarán los grupos.
- Cada grupo elaborará el proyecto y un informe. La entrega se realiza via assignmet en Gradescope. La entrega consta de: 1) El programa 2) Un informe en donde se explica las principales funciones del programa.
- 1) El programa debe tener terminada la implementación de las funcionalidades que indique su profesor
- 2) Informe debe incluir las siguientes secciones:
 - **Carátula:** Deberá incluir el título de su proyecto, el nombre del curso, el nombre de la universidad, el nombre y código de los integrantes del curso y el nombre del profesor.
 - **Resumen:** Deberá escribir un resumen de todas las secciones elaboradas hasta el momento. El resumen es la última sección en ser elaborada, pues contiene información de las demás secciones.
 - **Introducción** En esta sección, deberá hacer una breve introducción al tema de su proyecto. En esta sección deberá generar interés en el lector acerca de su tema. Puede mencionar algunos antecedentes, algunos proyectos similares, algunas investigaciones existentes sobre el tema de su proyecto, algunas preguntas sobre este tema que aún no han sido resueltas, o exponer algunos aspectos interesantes de su tema.
 - **Definiciones** En esta sección deberá describir los principales actores de su programa (en la siguiente sección, usted explicará el funcionamiento de su programa, detallando cómo interactúan dichos actores, por lo que es importante primero presentar una descripción detallada de cada actor). Esta sección servirá como un breve diccionario. Como recomendación se puede crear un diagrama de clases para saber como se va interactuar en el aplicativo.
 - **Funcionamiento del programa** Deberá mostrar el funcionamiento de su programa, incluyendo porciones de su código. En esta sección puede incluir algunas capturas de pantalla de las salidas de su programa si es que esto enriquece la explicación de su funcionamiento.
 - **Ejemplos de Ejecución** Deberá incluir ejemplos de ejecución de su programa en funcionamiento, tal y como se producen en pycharm o repl.it, tome de referencia los ejemplos de ejecución que se le muestran en las prácticas calificadas.
 - **Conclusiones y recomendaciones** Deberá presentar como mínimo 4 conclusiones y 4 recomendaciones relacionadas a su proyecto.

El informe deberá ser entregado en formato .pdf y deberá ser cargado a la misma tarea de Gradescope en la que entregó su programa. El archivo debe llamarse "Informe.PDF".

- El programa se sube via gradescope hasta el **día 19 de Noviembre antes de las 20 hrs.** Una entrega por grupo. No olvide incluir el archivo "Informe.PDF"
- **La Exposición grupal** se realiza el **día de la Sesión 15B según su horario**, durante la hora de clase.

- La exposición consta de dos partes:
- Primera parte el grupo hará una demostración de las funcionalidades del programa, siguiendo un guión basado en casos de prueba que evidencien el funcionamiento de cada una de las funcionalidades requeridas.
- Luego el profesor realizará las preguntas que considere convenientes, ya sea referidas a las funcionalidades o a la forma cómo se implemento el código.
- De acuerdo al desenvolvimiento de los integrantes del grupo y los ítem que se consignan en la rúbrica se define la nota.

Adicionalmente, tome en cuenta las siguientes consideraciones:

- La asistencia a la presentación es mandatoria para poder considerar la nota. Si uno de los participantes no asiste a la presentación, no tendrá nota. Exceptuando faltas justificadas a través de counter alumnos, bienestar estudiantil o directamente al profesor con un motivo de fuerza mayor (ej. Salud con descanso médico).
- Si el grupo no está completo y no tiene problemas al realizar la presentación del trabajo, podrá hacerlo, haciendo la salvedad si los integrantes faltantes trabajaron en equipo o no.
- Si por alguna razón, el grupo no puede presentar el proyecto dado que el código lo tiene alguien del equipo que no está presente, se considerará como una proyecto no presentado.
- Tome en cuenta que no hay recuperación sobre esta entrega, debe planificar con tiempo la entrega de esta parte.
- Si durante las preguntas, alguien del equipo no conoce el código presentado, la evaluación de todo el grupo se verá perjudicada. No es admisible decir 'Esa parte la hizo mi compañero'

3 CRITERIOS DE EVALUACIÓN

Criterio	Excelente (5pts)	Adecuado (4pts)	Mínimo (2pts)	Insuficiente (0pts)
Programación (5 pts)	Selecciona las estructuras de datos adecuadas e implementa los algoritmos necesarios para satisfacer el 100 % de los requerimientos técnicos del programa.	Selecciona las estructuras de datos adecuadas e implementa los algoritmos necesarios para satisfacer al menos el 80 % de los requerimientos técnicos del programa.	Selecciona las estructuras de datos adecuadas e implementa los algoritmos necesarios para satisfacer al menos el 65 % de los requerimientos técnicos del programa.	Selecciona las estructuras de datos adecuada e implementa menos del 65 % de los requerimientos técnicos del programa.
Funcionamiento (5 pts)	El proyecto cumple con el 100% de las funcionalidades esperadas y sin errores	El proyecto cumple con al menos el 80% de las funcionalidades esperadas y no presenta errores que alteren significativamente el resultado.	El proyecto cumple con al menos el 65% de las funcionalidades esperadas. El programa presenta errores que afectan el resultado.	El proyecto cumple con menos del 65% de las funcionalidades esperadas. El programa presenta errores que afectan el resultado de manera significativa.
Legibilidad y documentación (5pts)	El código del proyecto está distribuido en al menos dos módulos. Mas del 90% de código está documentado y es legible.	El código del proyecto está distribuido en al menos dos módulos. Mas del 80% de código está documentado y es legible.	El código del proyecto está distribuido en al menos dos módulos. Mas del 65% de código está documentado y es legible.	El código del proyecto no está distribuido en al menos dos módulos. Menos del 50% de código está documentado y no es legible.
Presentación (5pts)	Los principales algoritmos utilizados, son explicados de manera clara y precisa. Contestan al 100% de las preguntas planteadas y reconocen las ventajas y limitaciones de las herramientas y recursos utilizados.	Los principales algoritmos utilizados, son explicados de manera clara y precisa. Contestan al 80% de las preguntas planteadas y reconocen las ventajas y limitaciones de las herramientas y recursos utilizados.	Los principales algoritmos utilizados, son explicados de manera clara y precisa. Contestan al 65% de las preguntas planteadas y reconocen las ventajas y limitaciones de las herramientas y recursos utilizados.	Los principales algoritmos utilizados, no son explicados de manera clara y precisa. Contestan solo al 50% de las preguntas planteadas y no reconocen las ventajas y limitaciones de las herramientas y recursos utilizados.