

Gym en C

Totalmente desarrollada en C haciendo uso de todo los conceptos aprendidos durante el año así como lo son vectores, funciones, registros, archivos entre otros conceptos y buenas prácticas.





TFI - AyED

EQUIPO DE TRABAJO

¿Quiénes somos?



Ignacio Rosconi
Developer C



Ignacio Paez
Developer C



Francisco Perez
Scrum Master



Angel Pastrana
Developer C



Esteban Gerez
Developer C

Sistema para un Gimnasio programado en "C"

El lenguaje C es un lenguaje de programación de propósito general creado en 1972 por Dennis Ritchie en los laboratorios Bell de AT&T. Es un lenguaje de programación de nivel medio, lo que significa que combina características de lenguajes de bajo nivel (como la manipulación directa de la memoria del sistema) con características de lenguajes de alto nivel (como la estructuración del código en bloques funcionales).

Ha sido ampliamente utilizado en la industria del software debido a su eficiencia, flexibilidad y portabilidad. Una de las principales razones para trabajar con C es que permite un mayor control sobre los recursos del sistema, lo que es útil en aplicaciones de tiempo real y en sistemas operativos, entre otras áreas. Además, es muy versátil, lo que significa que puede ser utilizado en una gran variedad de aplicaciones, como la programación de sistemas operativos, la programación de microcontroladores, la creación de aplicaciones de escritorio y de dispositivos móviles, entre otros.

Además, es un lenguaje altamente portable, lo que significa que el código escrito en C se puede ejecutar en una amplia variedad de plataformas y sistemas operativos sin necesidad de cambios significativos.



Realizar un trabajo que integre todos los conocimientos de programación en C que se vieron durante un año en la facultad es beneficioso porque permite consolidar los conceptos y habilidades adquiridos a lo largo del curso, y aplicarlos en un proyecto práctico que puede ser de utilidad en el mundo real.

Además, esto proporciona una experiencia valiosa en la creación de software desde cero, que puede ser utilizada en futuros proyectos o en el desarrollo de una carrera profesional en el campo de la programación. El uso de archivos en C permite la manipulación de datos almacenados en disco, lo que es útil en una amplia variedad de aplicaciones, desde la creación de bases de datos hasta la implementación de algoritmos de procesamiento de datos.



Github es una plataforma popular para el control de versiones y el trabajo en equipo en proyectos de programación. Es especialmente útil para proyectos en C, ya que permite mantener un historial completo de todas las versiones del código y colaborar en equipo de manera efectiva. También ofrece herramientas para la gestión de problemas y tareas.

Es importante utilizar Github para mantener el orden de los archivos, facilitar la colaboración entre los miembros del equipo, mantener un historial completo de versiones, facilitar la revisión de código y planificación de tareas, y detectar errores de manera más eficiente. Es recomendable trabajar en "branches" (ramas) separadas para evitar errores y hacer pruebas antes de unir el código al proyecto principal.

En nuestro caso, para mejorar nuestro proyecto, utilizamos Trello, una herramienta de gestión de proyectos que nos permitió organizar nuestras tareas y proyectos en tableros virtuales. Trello es fácil de usar, flexible y personalizable, lo que nos permitió trabajar de manera más efectiva, colaborar mejor y mantener un seguimiento claro del progreso.



Recursos

Parámetros para Abrir Archivos Binarios

Parámetros para Abrir Archivos de Texto

Función fseek de C

Colores en C

Emojis (Se los puede usar para documentación)

+ Añada una tarjeta

Preguntas

¿Qué significan los colores?

¿Cómo seleccionar una tarea?

¿Cómo crear una lista?

¿Cómo crear una tarjeta?

¿Cómo mover una lista o una tarjeta?

+ Añada una tarjeta

Para Hacer (Tarea Agarrada)

¡LEERME!

Menú del Sistema (Finalizado)

Iniciar Sesión

Cerrar aplicación

Registrar Usuario (Administrador)

Registrar Usuario (Entrenador)

+ Añada una tarjeta

Para Hacer (Tarea sin Agarrar)

Módulos (¡LEERME!)

+ Añada una tarjeta

Módulo del Gym

Inicio de sesión.

Visualizar listado de Socios y actividad que desarrolla

Listado de Socios y Actividades.

Registrar Rutinas de Gimnasia.

+ Añada una tarjeta

Módulo de Recepción

Registrar Actividades de Socios

Listado de Socios

Listado de fechas de pagos

+ Añada una tarjeta

Para Hacer (Tarea Agarrada)

¡LEERME!

Menú del Sistema (Finalizado)

Iniciar Sesión

Cerrar aplicación

Registrar Usuario (Administrador)

Registrar Usuario (Entrenador)

+ Añada una tarjeta

Para Hacer (Tarea sin Agarrar)

Módulos (¡LEERME!)

+ Añada una tarjeta

Módulo del Gym

Inicio de sesión.

Visualizar listado de Socios y actividad que desarrolla

Listado de Socios y Actividades.

Registrar Rutinas de Gimnasia.

+ Añada una tarjeta

Módulo de Recepción

Registrar Actividades de Socios

Listado de Socios

Listado de fechas de pagos

+ Añada una tarjeta

Módulo de Administración

Registrar Actividades del Gym

Calcular pago del entrenador

Entrenador con mayor carga horaria

+ Añada una tarjeta

Blocked

+ Añada una tarjeta

¿Qué es el scrum master?

El Scrum Master es un miembro del equipo Scrum, que es un marco de trabajo ágil utilizado para desarrollar productos y proyectos de manera colaborativa. El Scrum Master es responsable de asegurar que el equipo Scrum esté siguiendo el marco de trabajo y cumpliendo con sus objetivos, y actúa como un facilitador, mentor y defensor del equipo Scrum. Además, el Scrum Master ayuda a eliminar obstáculos que impiden el progreso del equipo y promueve la colaboración entre los miembros del equipo.

¿Por qué Francisco es el Scrum Master?

En el equipo de trabajo se decidió que Francisco era la persona más adecuada para ser quien tenga esta posición de Scrum Master para poder ir organizando todo lo respectivo al Trello y todo el trabajo de las personas que pertenecen al grupo.

¿Qué técnicas se utilizó para desarrollar el programa?

Utilizamos lo que en nuestro grupo lo conocemos como la metodología “Pala” que es una forma de indicar de que nos pusimos entre el equipo de trabajo a hacer alguna reunión para avanzar con el TFI.



Con el equipo de trabajo tenemos un grupo de WhatsApp donde mandábamos este sticker de WhatsApp cada vez que necesitamos realizar algún tipo de reunión para avanzar en el proyecto.

¿Cómo funciona el sistema de control de versiones?

El sistema de control de versiones solamente fue administrado por el Scrum Master de manera que él fue el único que realizo commits y merge en el archivo principal,

	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
ultimo commit	#14 by FranprzDev was merged 36 minutes ago						
commit prefinal	#13 by FranprzDev was merged yesterday						
modulo 3 terminado	#12 by FranprzDev was merged last week						
Branch perez	#11 by FranprzDev was merged 2 weeks ago						
Corrección de errores.	#10 by FranprzDev was merged 2 weeks ago						
Programa con el switch principal 0 andando	#9 by FranprzDev was merged 2 weeks ago						
main.cpp & libreria.h	#8 by FranprzDev was merged 2 weeks ago						
Correción de Errores	#7 by FranprzDev was merged 2 weeks ago						
Merge BranchPerez to MainBranch	#6 by FranprzDev was merged 3 weeks ago						
Funcionando sin errores.	#5 by FranprzDev was merged last month						
Se agrego la funcionalidad del Login	#4 by FranprzDev was merged last month						
Register Andando...	#3 by FranprzDev was merged last month						
Registrar Usuario _ NORMAL _	#2 by FranprzDev was merged last month						
Menú del Sistema	#1 by FranprzDev was merged on Jan 16						

Como se ve, no existe una correlatividad en los nombres debido a que no nos pusimos de acuerdo e íbamos uniendo las ramas a medida que necesitábamos añadir un gran cambio. Esto se debe principalmente a un error de la primera vez que utilizamos GitHub en un ambiente colaborativo y más profesional.

Entonces, podemos concluir que entre la planificación y programación de nuestro Trabajo Final Integrador fue de aproximadamente de 1 mes en el tiempo que nos tomó realizarlo, pero no obstante se deja en claro que no se trabajó todos los días si no cuando el grupo podía ya que hubo bastantes problemas en cuestiones de tiempo debido a otros estudios formales que tomo el equipo de trabajo durante el mes de enero.



DOCUMENTACIÓN DEL CÓDIGO...

Aprende sobre cómo y de que
manera está hecho el código en
nuestro trabajo...





MENÚS

TODOS LOS MENÚS QUE SE UTILIZARON EN LA CODIFICACIÓN SON FUNCIONES DE TIPO ENTERAS, YA QUE AHÍ TAMBIÉN SE INGRESA LA OPCIÓN SOBRE LA QUE SE DESEA REALIZAR UNA ACCIÓN.

INT INGRESAR()

INT MENUPRINCIPAL()

INT MENUGYM()

INT MENURECEPCION()

INT SUBMENURECEPCION()

INT MENUJOINGYM()

INT MENUADMIN()

INT TYPEUSER()

VOID CASODIEZ()



REGISTER

SON FUNCIONES ENFOCADAS AL REGISTRO DE UN USUARIO EN PARTICULAR (ADMINISTRADOR/ENTRENADOR)

VOID GENERARCUENTAADMINISTRADOR(FILE)
GENERA LA CUENTA ADMINSTRADOR

VOID GENERARCUENTAENENTRENADORES(FILE)
GENERA LA CUENTA ENTRENADOR

INT VALIDARUNICIDADADMIN(CHAR [10], FILE)
VALIDA SI EL ADMIN ES ÚNICO EN EL ARCHIVO.

INT VALIDARCOM(CHAR [10])
VALIDA SI COMIENZA EN MINUSCULA

INT VALIDARMAYUSCULA(CHAR[10])
BUSCA ALMENOS 2 MAYUSCULAS EN EL NOMBRE DE USUARIO

INT VALIDARMINIMO(CHAR[10])
BUSCA SI HAY COMO MÁXIMO 3 NÚMEROS EN EL NOMBRE ED USUARIO.

INT VALIDARNOMBREUSUARIOADMIN(CHAR[10], FILE)
ES LA FUNCIÓN QUE LLAMARÁ A LAS FUNCIONES DE VALIDACIÓN.

INT VALIDARUNICIDADENTRENADOR(CHAR[10], FILE)
VALIDA SI EL ENTRENADOR ES ÚNICO EN EL ARCHIVO.



REGISTER

SON FUNCIONES ENFOCADAS AL REGISTRO DE UN USUARIO EN PARTICULAR (ADMINISTRADOR/ENTRENADOR)

INT VALIDARNOMBREUSUARIOENTRENADOR(CHAR[10], FILE)
ES LA FUNCIÓN QUE LLAMARÁ A LAS FUNCIONES DE VALIDACIÓN

INT VALIDARCONTRASENIA(CHAR[32], INT)
ES LA FUNCIÓN QUE LLAMARÁ A LAS FUNCIONES DE VALIDACIÓN DE CONTRASENIA
DICHAS FUNCIONES SON COMPARTIDAS...

INT CALCLONG(CHAR[32])
CALCULA LA LONGITUD DE LA CONTRASENIA.

INT VALIDTRIPLE(CHAR[32], INT)
VERIFICA SI EXISTEN NÚMEROS, MAYUSCULAS Y MINUSCULAS

INT CARACTALFA(CHAR[32], INT)
VALIDA SI LOS CARACTERES SON SOLAMENTE ALFANÚMERICOS.

INT CHARACTERCONSECNUM(CHAR[32], INT)
VALIDA SI EXISTEN CARACTERES CONSECUTIVOS NÚMERICOS.

INT CONSECUTIVALFAB(CHAR[32], INT)
VALIDA SI EXISTEN CARACTERES CONSECUTIVOS ALFÁBETICOS.



GYM

SON FUNCIONES ENFOCADAS AL USO DEL MODULO DEL GIMNASIO CON LAS FUNCIONALIDADES PROPEUSTAS.

INT LOGUEARENTRENAORENDB(FILE, CHAR[10])
LLAMA A LAS FUNCIONES ALEDAÑAS PARA EL LOGIN.

INT VALIDARUSUARIOENTRENADOR(FILE, CHAR[32], CHAR[10])
VALIDA SI EL ENTRENADOR ES VÁLIDO (FUNCIÓN DE LOGIN)

INT LOGUEARENTRENADOR(FILE, CHAR[10])
LLAMA AL INGRESO PARA EL LOGIN (USUARIO & CONTRASENIA)

INT VALIDARUSUARIOENTRENADOR(FILE, CHAR[32], CHAR[10])
VALIDA QUE EL USUARIO EXISTA EN NUESTRO ARCHIVO,
EN CASO DE QUE SEA SI DEVUELVE 1.

VOID LISTADOSOCIOSACTIVIDAD(FILE)
LISTA LA ACTIVIDAD DE LOS SOCIOS A PARTIR DE UN ARCHIVO.

VOID REGISTRARRUTINA(FILE)
REGISTRA UNA RUTINA ESPECIFICA EN EL SOCIO.



RECEPCIÓN

SON FUNCIONES ENFOCADAS AL USO DEL MODULO DE RECEPCIÓN CON LAS FUNCIONALIDADES PROPEUSTAS.

INT LOGUEARADMINISTRADOR(FILE, CHAR[10])
LLAMA A LAS FUNCIONES ALEDAÑAS PARA EL LOGIN, AFIRMATIVO
DEVUELVE 1.

INT LOGUEARADMINISTRADOR(FILE, CHAR[10])
LLAMA AL INGRESO PARA EL LOGIN (USUARIO & CONTRASEÑA)

INT VALIDARADMIN(FILE, CHAR[32], CHAR[10])
VALIDA SI EL ADMINISTRADOR EXISTE EN EL ARCHIVO EN CASO
AFIRMATIVO DEVUELVE 1.

VOID REGISTRARSOCIO(FILE, INT, CHAR[10])
REGISTRA UN SOCIO EN EL ARCHIVO

INT GENERARNUMSOC()
GENERA UN NÚMERO DE SOCIO PARA EL SOCIO POR REGISTRAR
(INTERNA A REGISTRARSOCIO)

VOID LISTARSOCIOS(FILE)
LISTA LOS SOCIOS QUE SE ENCUENTRAN EN EL ARCHIVO.

VOID REGISTRARACTIVIDADSOCIO(FILE, FILE, INT)
BUSCA UN SOCIO Y REGISTRA LA ACTIVIDAD SEGÚN SU NÚMERO DE
SOCIO.

VOID LISTARFECHAPAGOS(FILE)
LISTA LA FECHA DE PAGOS DE LOS SOCIO



ADMINISTRACIÓN

SON FUNCIONES ENFOCADAS AL USO DEL MODULO DE ADMINISTRACIÓN LAS FUNCIONALIDADES PROPEUSTAS.

INT LOGUEARADMINISTRADORDB(FILE, CHAR[10])
LLAMA A LAS FUNCIONES ALEDAÑAS PARA EL LOGIN, AFIRMATIVO
DEVUELVE 1.

INT LOGUEARADMINISTRADOR(FILE, CHAR[10])
LLAMA AL INGRESO PARA EL LOGIN (USUARIO & CONTRASENIA)

INT VALIDARADMIN(FILE, CHAR[32], CHAR[10])
VALIDA SI EL ADMINISTRADOR EXISTE EN EL ARCHIVO EN CASO
AFIRMATIVO DEVUELVE 1.

VOID REGISTRARACTIVIDAD(FILE, FILE, FILE, CHAR[10])
REGISTRA UNA ACTIVIDAD EN EL ARCHIVO "ARCHACTIVIDADES"

INT ENCONTRARENTRENADOR(FILE, INT)
ENCUENTRA POR LEGAJOS UN ENTRENADOR EN PARTICULAR.
DEVUELVE 1 EN CASO AFIRMATIVO.

INT COMPROBARACTIVIDADES(FILE, INT, INT, INT, STRUCT ACTIVIDADES)
COMPRUEBA SI NO COINCIDEN LAS ACTIVIDADES
EN MISMOS DÍAS Y TURNOS.
DEVUELVE 1 SI ENCONTRÓ UNA ACTIVIDAD EN EL MISMO TURNO.



TFI - AYED

ADMINISTRACIÓN

SON FUNCIONES ENFOCADAS AL USO DEL MODULO DE ADMINISTRACIÓN LAS FUNCIONALIDES PROPEUSTAS.

INT COMPROBARTURNO(FILE, STRUCT ACTIVIDADES, TURNO)
COMPRUEBA SI EL TURNO ESTÁ DISPONIBLE

INT GENERARCODIGOACTIVIDAD()
GENERA UN CODIGO DE ACTIVIDAD ENTRE [1000000000,10000000000]

VOID CALCULARPAGOENTRENADOR(FILE, INT)
CALCULA EL PAGO DE UN ENTRENADOR EN PARTICULAR, MEDIANTE
SU LEGAJO.

VOID CALCULARMAYORCARGAHORARIA(FILE, FILE)
CALCULA LA MAYOR CARGA HORARIA DE LOS ENTRENADORES.



conclusion.cpp

```
#include<stdio.h>
```

```
main(){  
printf("Como conclusión  
podemos decir que el trabajo  
fue bastante enriquecedor y  
nos proporciono las  
herramientas necesarias para  
poder entender más a fondo el  
manejo de C y la programación  
imperativa.");
```

```
printf("\n\nMuchas gracias  
por este proyecto\n\n");  
  
}
```

Trabajo Final Integrador - Gimnasio en C