

Universidad de San Carlos de Guatemala
Laboratorio de Estructura de Datos
Auxiliar: Carlos Javier Castro González

Fase #2 Proyecto Socia Structure

Angel Guillermo de Jesús Pérez Jiménez
202100215
Guatemala 21 de septiembre de 2024

Objetivo general

Desarrollar una plataforma social que permita a los usuarios interactuar, compartir publicaciones y gestionar relaciones, utilizando estructuras de datos eficientes para almacenar y procesar la información de manera óptima.

Objetivos específicos

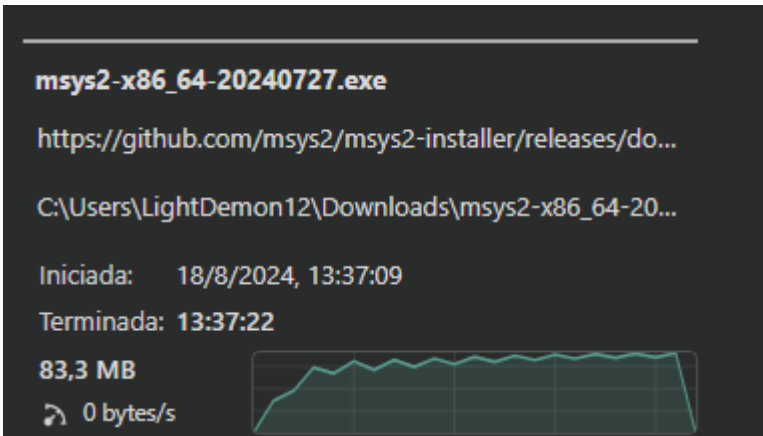
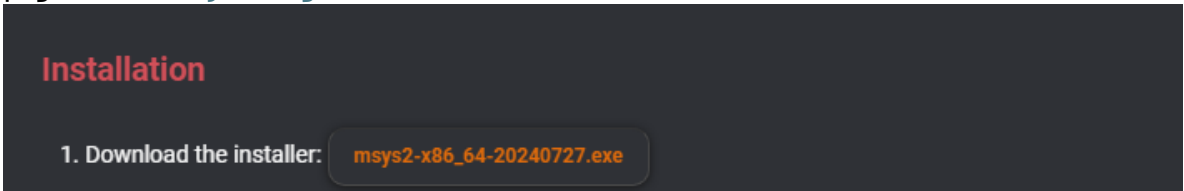
Implementar un sistema de gestión de usuarios que permita el registro, inicio de sesión y manejo de perfiles, incluyendo la capacidad de establecer y gestionar amistades y relaciones con otros usuarios.

Desarrollar una interfaz gráfica intuitiva que facilite la interacción de los usuarios con la plataforma, permitiendo la creación, visualización y gestión de publicaciones, así como la visualización de comentarios y reacciones.

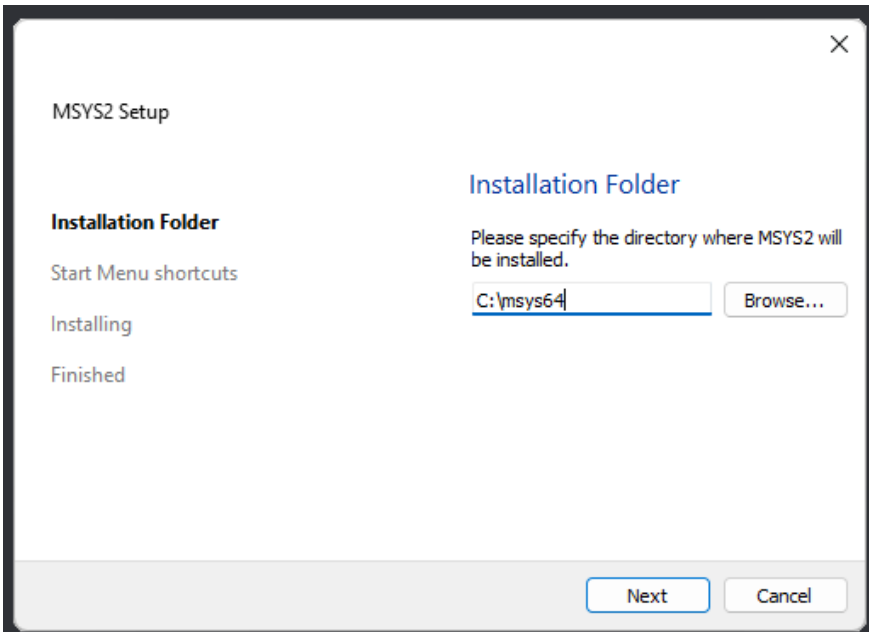
Optimizar el almacenamiento y recuperación de datos mediante el uso de estructuras de datos como árboles AVL y listas dobles, garantizando un rendimiento eficiente en la gestión de grandes volúmenes de información y mejorando la experiencia del usuario en la plataforma.

Instalación de C++ en el sistema

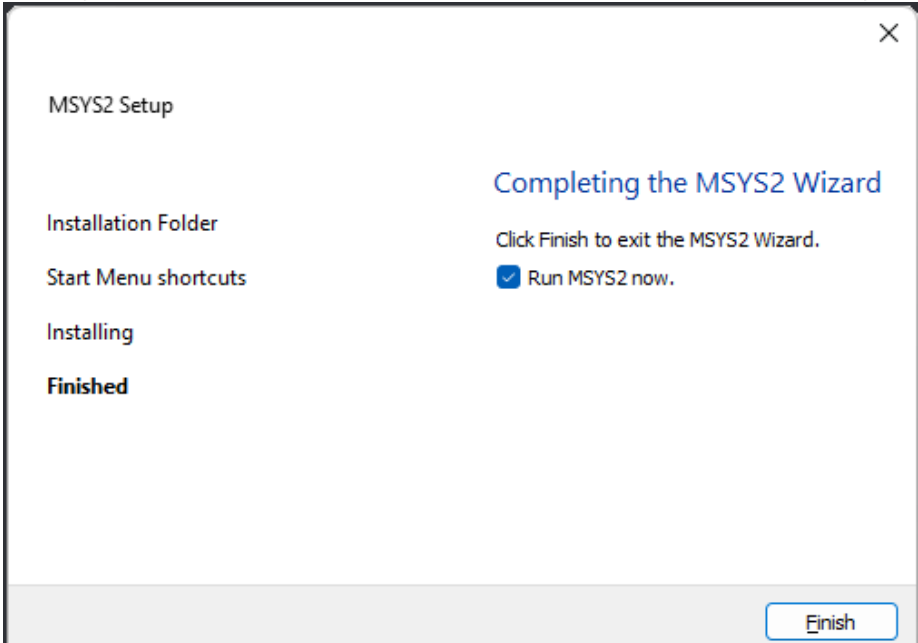
- 1. Se descargo MSYS2 para la instalación del compilador de c/c++ desde su pagina www.msys2.org



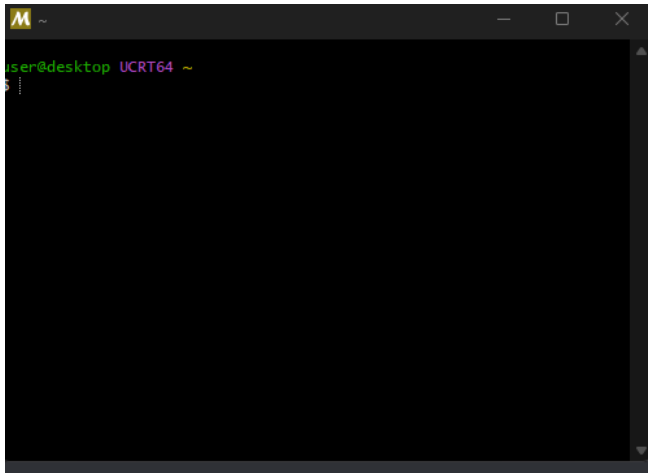
- 2. Una vez descargado el ejecutable se procedió a ejecutarlo para seguir con la instalación



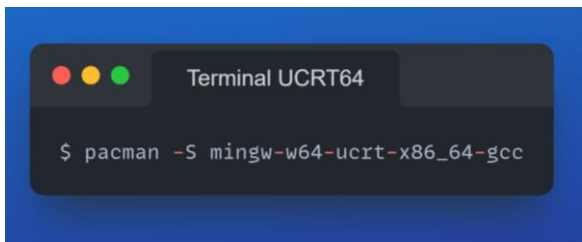
- 3. Se siguieron los pasos del instalador sin modificar ningún parámetro



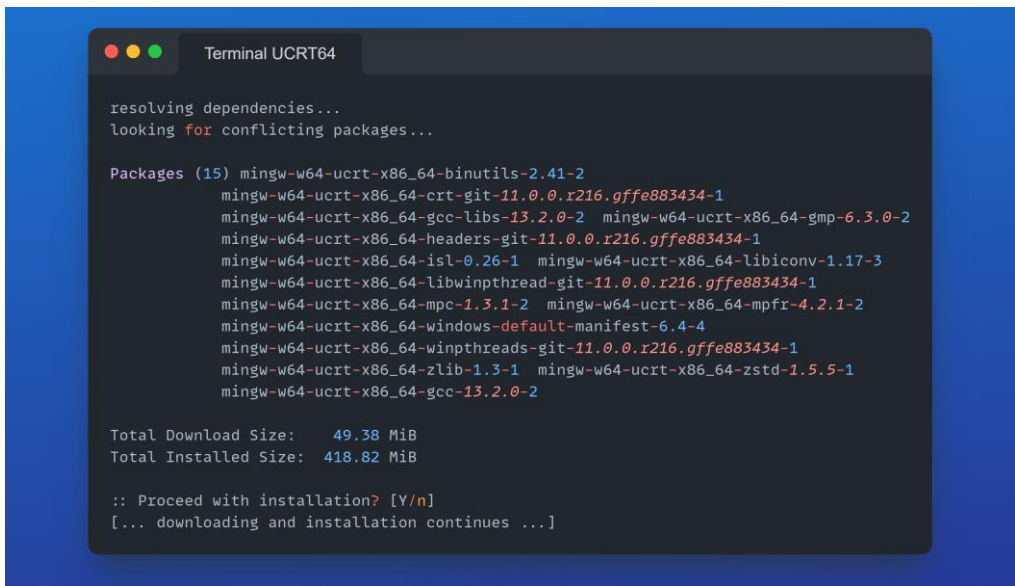
4. Una vez instalador se lanzo la terminal de UCRT64



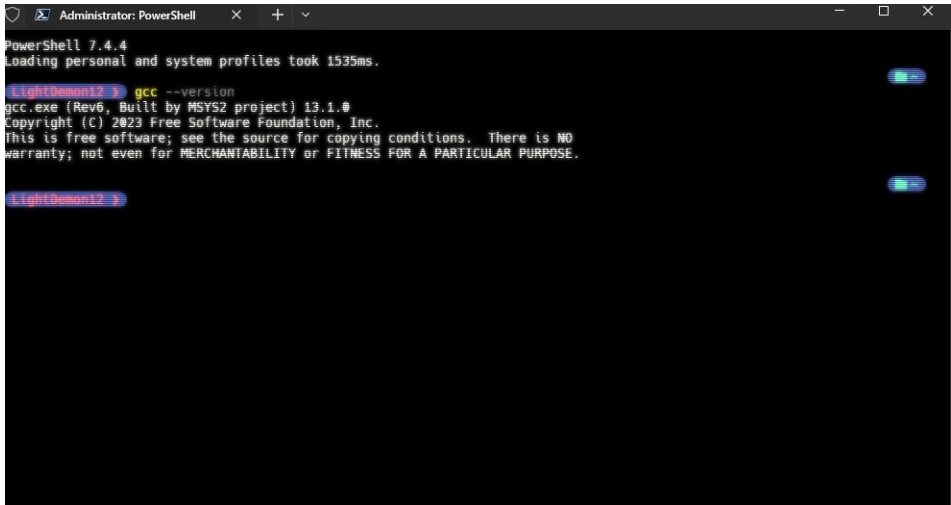
5. Dentro de esta terminal se ejecuto el siguiente código para la instalación de otras herramientas



6. La terminal muestra una salida es necesario presionar ENTER para proseguir



7. Con el siguiente comando se puede comprobar que gcc está instalado en el sistema



Instalación de la librería nlohmann para la lectura de archivos JSON

1. Desde el apartado de Releases en el GITHUB de nlohmann se descargo el archivo json.hpp



2. Dentro del proyecto en Clion se creo una carpeta con el nombre nlohmann donde se incluyo el archivo json.hpp descargado anteriormente.



Estructura del proyecto



Estructuras de Datos

Usuario

Funciones Principales:

```
Usuario(const std::string& nombres, const std::string& apellidos, ...):  
Constructor que inicializa un nuevo usuario.  
void setNombres(const std::string& nombres): Establece el nombre del usuario.  
ListaEnviados& getListaEnviados(): Devuelve la lista de publicaciones enviadas  
por el usuario.
```

Amistades

Funciones Principales:

```
void agregarAmistad(const std::string& amigo): Agrega un nuevo amigo a la lista  
de amistades.  
void eliminarAmistad(const std::string& amigo): Elimina un amigo de la lista de  
amistades.  
void listarAmistades(): Muestra todas las amistades del usuario.
```

Relaciones

Funciones Principales:

```
void agregarRelación(const std::string& usuario): Agrega una nueva relación con  
otro usuario.  
void eliminarRelación(const std::string& usuario): Elimina una relación  
existente.  
void listarRelaciones(): Muestra todas las relaciones del usuario.
```

ListaDoble

Funciones Principales:

```
void insertar(const std::string& correo, const std::string& contenido, ...):  
Inserta una nueva publicación en la lista.  
void eliminar(const std::string& contenido): Elimina una publicación  
específica.  
void mostrarEnTabla(QTableWidget* tabla): Muestra las publicaciones en una  
tabla de la interfaz.
```

ArbolAVL

Funciones Principales:

```
void insertar(const std::string& valor): Inserta un nuevo valor en el árbol AVL  
y reequilibra si es necesario.  
void eliminar(const std::string& valor): Elimina un valor del árbol y  
reequilibra si es necesario.  
NodoAVL* buscar(const std::string& valor): Busca un nodo en el árbol.
```

ArbolBinarioCompleto

Funciones Principales:

```
void insertar(const std::string& valor): Inserta un nuevo nodo en el árbol  
binario completo.  
void mostrar(): Muestra todos los nodos del árbol en una estructura específica.  
void graficarArbol(const std::string& nombreArchivo): Genera una representación  
gráfica del árbol.
```

MatrizDispersa

Funciones Principales:

```
void insertar(int fila, int columna, double valor): Inserta un valor en la  
posición especificada de la matriz.  
void eliminar(int fila, int columna): Elimina el valor en la posición  
especificada.  
double acceder(int fila, int columna): Devuelve el valor en la posición  
especificada.
```

Pila

Funciones Principales:

```
void push(int valor): Agrega un valor a la parte superior de la pila.  
int pop(): Elimina y devuelve el valor en la parte superior de la pila.  
bool estaVacía(): Verifica si la pila está vacía.
```