

Calculadora con operandos en texto (Borrador 1)

Universidad de Costa Rica

Seguridad de Sistemas Computacionales

CI0143 - I Semestre 2025

Steven Castro Oreamuno

Rafa Isaac Gutiérrez Ramírez C03640

I-2025

Indice

1. [Descripcion](#)
2. [Uso](#)
3. [Descarga](#)
4. [Arquitectura](#)
 - [Version Plana](#)
 - [Version Estructurada](#)
5. [Comparacion Estruura;](#)
6. [Pruebas](#)

Descripcion

Programa de calculadora en base a texto:

- Acepta numeros del 0 al 99 en forma de texto y operandos basicos y parentesis:
- (, +, *, /, -)
- Realiza las operaciones aritmeticas
- Requiere la autenticacion del usuario
- Da autorizacion a los administradores de crear usuarios

Objetivos de seguridad :

Los siguientes objetivos de seguridad fueron los que se buscaba cumplir con la realizacion de esta tarea:

- Autenticacion de usuarios
- Autorizacion de usuarios de mayor nivel (administradores)

- Integridad de los operandos y los resultados
 - Confidencialidad de los credenciales(archivo texto)
 - Bitacora de actividades
-

Uso

Descarga

Se puede obtener ambas versiones mediante la clonacion del repositorio con el siguiente comando en la terminal:

```
git clone https://github.com/StevenCastroUCR/Seguridad.git
```

Compilacion

Usando la terminal desde la ubicacion del archivo fuente [/calcv0 para la version plana y /calcStrucutured para la version estructurada]laccorre el comando

```
make
```

Una vez compilado se debe correr con el comando

```
./bin/calculator
```

Arquitectura

Version Plana

- El programa y sus funciones se ubican en la clase main.cc
- Funciones:
- void menu(bool isAdmin);
- string textParser();
- int precedence(const std::string &op);
- int evalPostfix(const std::vector<std::string> &postfix);
- std::vector<std::string> infixToPostfix(const std::vector<std::string> &tokens);
- std::vector<std::string> tokenize(const std::string &expr);
- bool authentication(bool &isAdmin);
- string readInput(const string &prompt);
- void createUser();
- bool userExists(const string &userVerification);
- string xorCipher(const string &input);
- void log(const string &event);

Version estructurada

- **Modelo:** Clase Parser y User manejan la logica de calculadora y de usuarios
- **Vista:** El usuario introduce sus consultas y es dirigido mediante menus y opciones
- **Controlador:** La clase Controller maneja el flujo de la ejecucion

Clases

- **Parser:**
 - int precedence(const std::string &op); Funcion encargada de ver el orden de precedencia
 - std::vector<std::string> tokenize(const std::string &expr); Funcion que convierte en tokens la entrada del usuario para leerlos por separado
 - std::vector<std::string> infixToPostfix(const std::vector<std::string> &tokens); Funcion que convierte la notacion infija a posfija
 - string inputToNumbers(const string& inputToCheck); Funcion que convierte las entradas de letras en numeros
 - double evalPostfix(const std::vector<std::string>& postfix); Funcion que calcula el resultado desde la notacion posfija
 - bool opContinue(); Funcion que le pregunta al usuario si desea continuar
- **User:**
 - bool authentication(bool &isAdmin); Funcion encargada de autenticar a los usuraiois desde un documento de texto
 - void createUser(); Funcion encargada de crear un usuario
 - bool userExists(const std::string &userVerification); Funcion que verifica si el usuario existe
- **Encryption:**
 - void encrypt(); Funcion que encripta valores usando el cifrado XOR
 - string xorCipher(const string &input); Funcion de cifrado XOR
- **Controller:**
 - int run(); Funcion encargada de la ejecucion del programa
 - void Menu(bool isAdmin); Funcion que despliega menu de opciones a los usuarios
- **Utility:**
 - string readInput(const string &prompt); Funcion de lectura de entradas de los usuarios
 - void log(const string &event); Funcion de bitacora que guarda los eventos exitos y de error de los usuarios

Comparacion de Versiones segun Seguridad Estructural

| Categoria | Version 1 <i>Plana</i> | Version 2 <i>Estructurada</i> |
|---------------------------|------------------------------------|---|
| Organizacion de Codigo | Logica en un solo lugarmain.cpp | Division por Modelos, Vistas, Controladores, Interfaces y Utilidades |

| Categoria | Version 1 <i>Plana</i> | Version 2 <i>Estructurada</i> |
|-------------------------------|--|--|
| Complejidad | Alta - Responsabilidades revueltas | Reducidas - Responsabilidades separadas por Rol |
| Puntos de acceso de seguridad | Mezclados con una logica general | Aislados en casos dedicados (<i>User</i> , <i>Parser</i> , <i>Controller</i>) |
| Encapsulacion | Ninguna - Todos los datos y logica estan expuestos | Fuerte - Miembros de las clases encapsulan la logica y datos sensibles |
| Validacion | Minima - en linea | Centralizada por classes, reforzada antes de la ejecucion del codigo |
| Reutilizacion deCodigo | Baja - Funciones repetidas | High - clases y funciones de asistencia permiten la reutilizacion |
| Mantenibilidad | Dificil - Cambios pequeños rompen el flujo completo | Mas sencilla(necesita trabajo) - Los componentes son modulares y se pueden probar |
| Manejo de credenciales | Inseguro - riesgo de exposicion | Seguro (exposicion en la entrada requiere trabajo) - manejados en la clase <i>User</i> |
| Integridad de los datos | Debil - poco estructuracion de resultados y validaciones | Mas fuerte (requiere trabajo) - parser valida entradas |

Diagrama de Clases version Plana

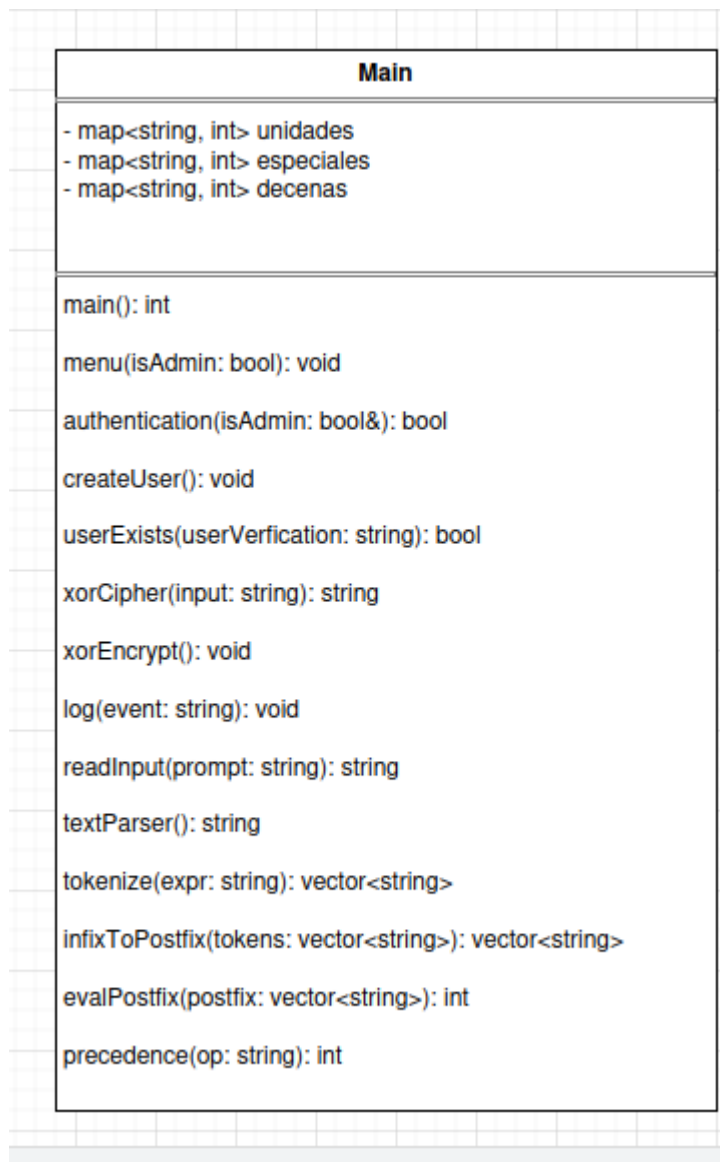
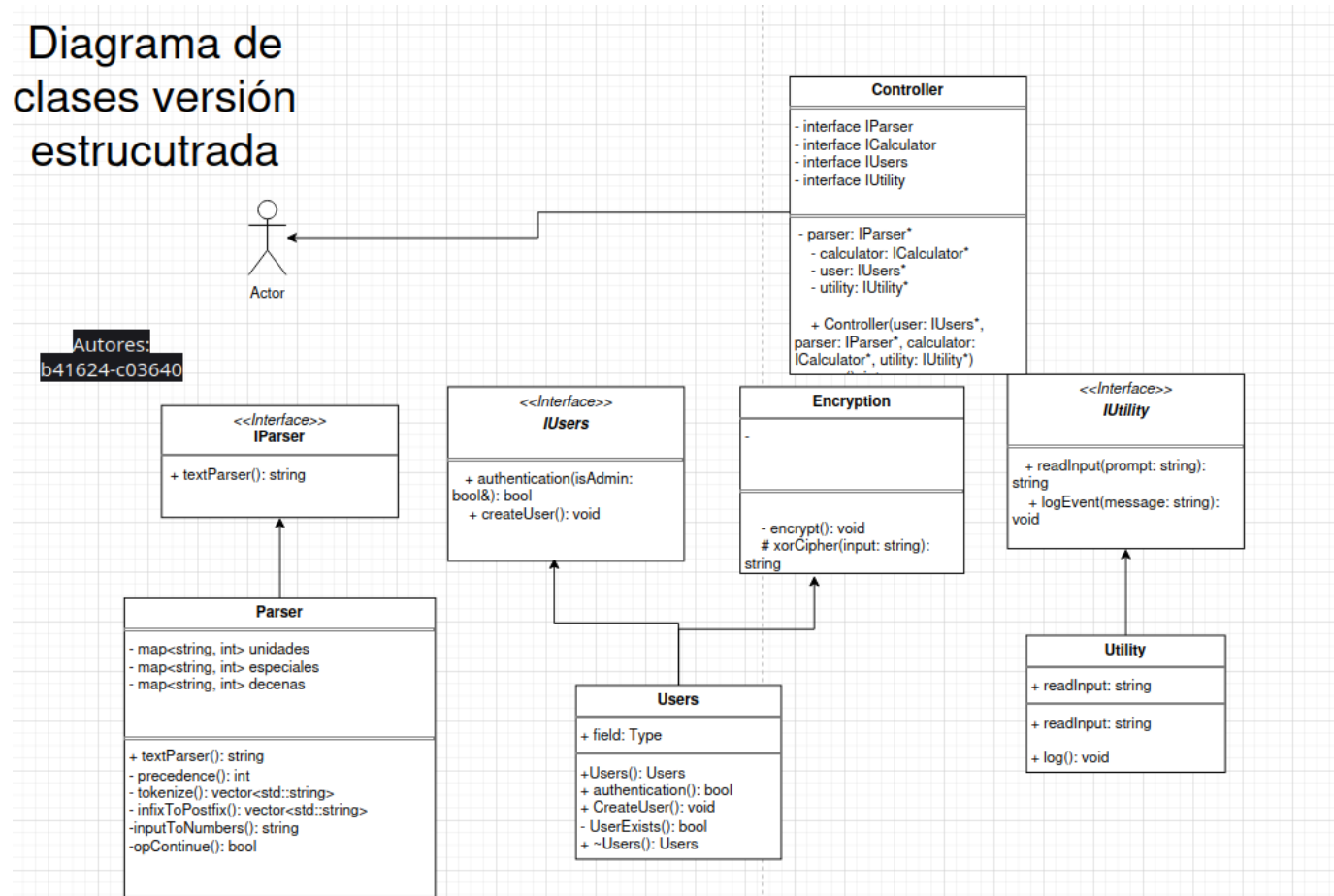


Diagrama de clases version Estructurada

Diagrama de clases versión estructurada



Imagenes de prueba Plana

```

obj/Encryption.o
• vboxuser@INDEXLAP:~/Documents/seg/Seguridad/calculStructured$ ./bin/calculator
Digite nombre usuario : Rafa
Digite la contraseña: 54321
Bienvenido: Rafa!
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion: 1
Escriba la operacion:
(cinco *(tres+dos)/cuatro)
Resultado: 6.25
Quiere hacer otra operacion? 1-) Si , 2-) No:
2
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion: 1
Escriba la operacion:
dos noventa y nueve
Escriba la operacion:
tres
Resultado: 3
Quiere hacer otra operacion? 1-) Si , 2-) No:
2
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion: 1
Escriba la operacion:
dos
Resultado: 2
Quiere hacer otra operacion? 1-) Si , 2-) No:
2
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion:

```

Crea usuario admin.

```

obj/Encryption.o
• vboxuser@INDEXLAP:~/Documents/seg/Seguridad/calculStructured$ ./bin/calculator
Digite nombre usuario : admin
Digite la contraseña: pass
Cuenta Administrador Bienvenido: admin!
Laboratorio 01 de seguridad
Calculadora texto

1. Crear usuario
2. Calcular de Texto
3. Salir
Digite una opcion: 1
Digite el nombre Usuario nuevo: felix
Digite la contraseña: 1
Digite la contraseña nuevamente: 1
Es un usuario administrador ? 1-) Si , 2-) No:
1
Usuario anadido correctamente
Laboratorio 01 de seguridad
Calculadora texto

1. Crear usuario
2. Calcular de Texto
3. Salir
Digite una opcion: 3
• vboxuser@INDEXLAP:~/Documents/seg/Seguridad/calculStructured$ ./bin/calculator
Digite nombre usuario : felix
Digite la contraseña: 1
Cuenta Administrador Bienvenido: felix!
Laboratorio 01 de seguridad
Calculadora texto

1. Crear usuario
2. Calcular de Texto
3. Salir
Digite una opcion: 1

```

Hace calculo.

Imágenes de prueba Estructurado

```
obj/Encryption.o
• vboxuser@INDEXLAP:~/Documents/seg/Seguridad/calculStructured$ ./bin/calculator
Digite nombre usuario : Rafa
Digite la contraseña: 54321
Bienvenido: Rafa!
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion: 1
Escriba la operacion:
(cinco *(tres+dos)/cuatro)
Resultado: 6.25
Quiere hacer otra operacion? 1-) Si , 2-) No:
2
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion: 1
Escriba la operacion:
dos noventa y nueve
Escriba la operacion:
tres
Resultado: 3
Quiere hacer otra operacion? 1-) Si , 2-) No:
2
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion: 1
Escriba la operacion:
dos
Resultado: 2
Quiere hacer otra operacion? 1-) Si , 2-) No:
2
Laboratorio 01 de seguridad
Calculadora texto

1. Calcular de Texto
2. Salir
Digite una opcion:
```

Crea nuevo usuario

```
obj/Encryption.o
• vboxuser@INDEXLAP:~/Documents/seg/Seguridad/calculStructured$ ./bin/calculator
Digite nombre usuario : admin
Digite la contraseña: pass
Cuenta Administrador Bienvenido: admin!
Laboratorio 01 de seguridad
Calculadora texto

1. Crear usuario
2. Calcular de Texto
3. Salir
Digite una opcion: 1
Digite el nombre Usuario nuevo: felix
Digite la contraseña: 1
Digite la contraseña nuevamente: 1
Es un usuario administrador ? 1-) Si , 2-) No:
1
Usuario anadido correctamente
Laboratorio 01 de seguridad
Calculadora texto

1. Crear usuario
2. Calcular de Texto
3. Salir
Digite una opcion: 3
• vboxuser@INDEXLAP:~/Documents/seg/Seguridad/calculStructured$ ./bin/calculator
Digite nombre usuario : felix
Digite la contraseña: 1
Cuenta Administrador Bienvenido: felix!
Laboratorio 01 de seguridad
Calculadora texto

1. Crear usuario
2. Calcular de Texto
3. Salir
Digite una opcion: 1
```

Realiza calculo