

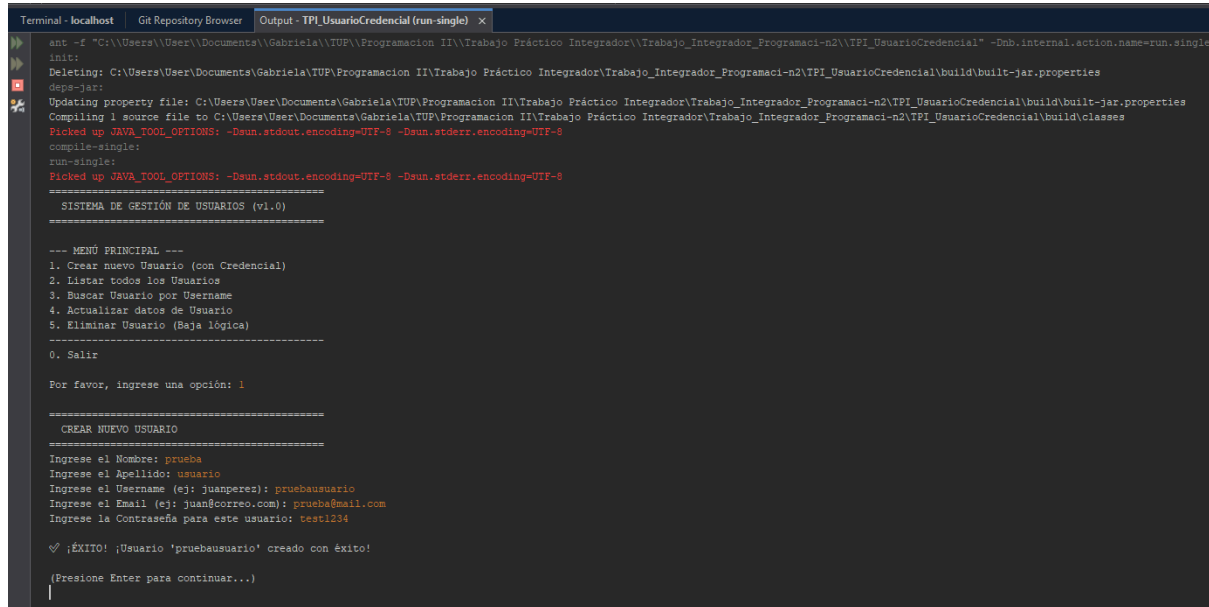
Reporte de casos de prueba manuales ejecutados

Categorías cubiertas:

- ✓ Validaciones de entrada e integridad (CP01, CP02, CP03, CP04, CP05, CP12)
- ✓ Operaciones CRUD completas (CP06, CP07, CP08, CP09, CP10, CP11)
- ✓ Reglas de negocio y manejo de errores (CP02, CP03, CP04, CP05, CP06, CP08, CP12)

CP01 - Creación Exitosa de Usuario con Credencial

Descripción: Verificar que se puede crear un usuario correctamente con todas las validaciones pasando



```
Terminal - localhost  Git Repository Browser  Output - TPI_UsuarioCredencial (run-single) x
ant -f "C:\Users\User\Documents\Gabriela\TUP\Programacion II\Trabajo Práctico Integrador\Trabajo_Integrador_Programaci-n2\TPI_UsuarioCredencial" -Dnb.internal.action.name=run.single
init:
Deleting: C:\Users\User\Documents\Gabriela\TUP\Programacion II\Trabajo Práctico Integrador\Trabajo_Integrador_Programaci-n2\TPI_UsuarioCredencial\build\build-jar.properties
deps-jar:
Updating property file: C:\Users\User\Documents\Gabriela\TUP\Programacion II\Trabajo Práctico Integrador\Trabajo_Integrador_Programaci-n2\TPI_UsuarioCredencial\build\build-jar.properties
Compiling 1 source file to C:\Users\User\Documents\Gabriela\TUP\Programacion II\Trabajo Práctico Integrador\Trabajo_Integrador_Programaci-n2\TPI_UsuarioCredencial\build\classes
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
compile-single:
run-single:
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
=====
SISTEMA DE GESTIÓN DE USUARIOS (v1.0)
=====

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

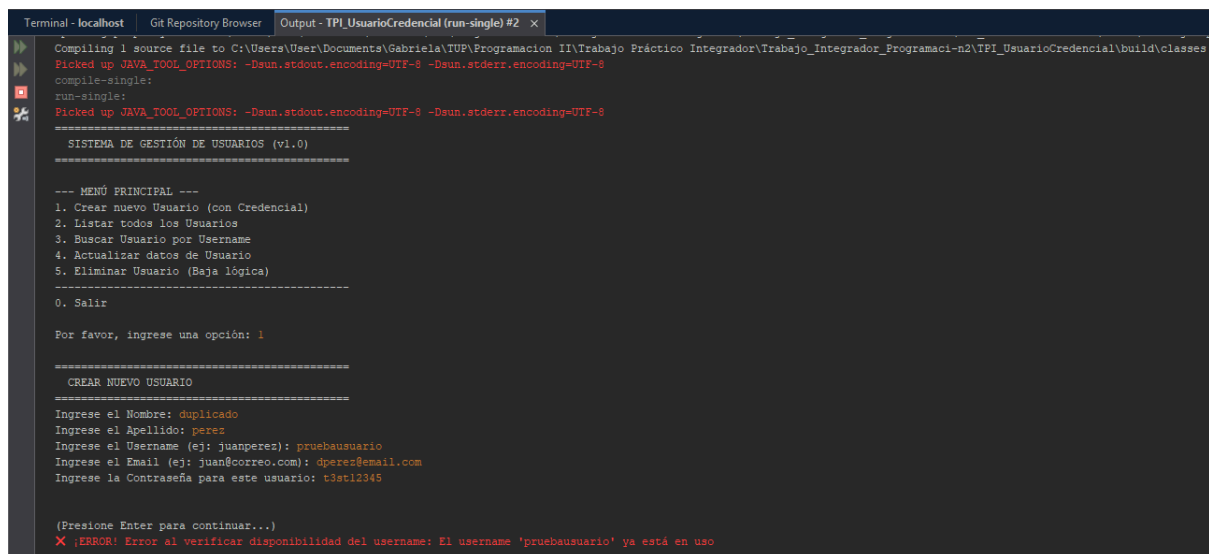
=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: prueba
Ingrese el Apellido: usuario
Ingrese el Username (ej: juanperez): pruebausuario
Ingrese el Email (ej: juan@correo.com): prueba@mail.com
Ingrese la Contraseña para este usuario: test1234

✓ ¡ÉXITO! ¡Usuario 'pruebausuario' creado con éxito!

(Presione Enter para continuar...)
```

CP02 - Validación de Username Duplicado

Descripción: Verificar que no se permiten usernames duplicados (se ingresó el id del usuario creado en el caso anterior).



```
Terminal - localhost  Git Repository Browser  Output - TPI_UsuarioCredencial (run-single) #2 x
Compiling 1 source file to C:\Users\User\Documents\Gabriela\TUP\Programacion II\Trabajo Práctico Integrador\Trabajo_Integrador_Programaci-n2\TPI_UsuarioCredencial\build\classes
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
compile-single:
run-single:
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
=====
SISTEMA DE GESTIÓN DE USUARIOS (v1.0)
=====

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: duplicado
Ingrese el Apellido: perez
Ingrese el Username (ej: juanperez): pruebausuario
Ingrese el Email (ej: juan@correo.com): qperez@mail.com
Ingrese la Contraseña para este usuario: t3st12345

(Presione Enter para continuar...)
✗ ¡ERROR! Error al verificar disponibilidad del username: El username 'pruebausuario' ya está en uso
```

CP03 - Validación de Email Inválido

Descripción: Verificar validación de formato de email

```
Terminal - localhost  Git Repository Browser  Output - TPI_UsuarioCredencial (run-single) #2  X

Ingrese el Username (ej: juanperez): pruebausuario
Ingrese el Email (ej: juan@correo.com): dperez@email.com
Ingrese la Contraseña para este usuario: t3st12345

(Presione Enter para continuar...)
X ¡ERROR! Error al verificar disponibilidad del username: El username 'pruebausuario' ya está en uso

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Martin
Ingrese el Apellido: Juárez
Ingrese el Username (ej: juanperez): mjuarez
Ingrese el Email (ej: juan@correo.com): martin@
Ingrese la Contraseña para este usuario: artdhcyi874

X ¡ERROR! El email es obligatorio y debe tener formato válido (ej: usuario@dominio.com)
(Presione Enter para continuar...)
```

CP04 - Validación de Campos Obligatorios

Descripción: Verificar que campos requeridos no pueden estar vacíos (nombre, apellido, username, email y contraseña)

```
Terminal - localhost  Git Repository Browser  Output x
TPI_UsuarioCredencial (run-single) #2 x  TPI_UsuarioCredencial (run-single) #3 x

compile-single:
run-single:
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
=====
      SISTEMA DE GESTIÓN DE USUARIOS (v1.0)
=====

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
      CREAR NUEVO USUARIO
=====

Ingrese el Nombre:
Ingrese el Apellido: Hernández
Ingrese el Username (ej: juanperez): t.hernandez
Ingrese el Email (ej: juan@correo.com): thernandez@corp.com
Ingrese la Contraseña para este usuario: t34789op

(Presione Enter para continuar...)
X ¡ERROR! El nombre del usuario es obligatorio
```

```
TPI_UsuarioCredencial (run-single) #2 x  TPI_UsuarioCredencial (run-single) #3 x

(Presione Enter para continuar...)
X ¡ERROR! El nombre del usuario es obligatorio

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
      CREAR NUEVO USUARIO
=====

Ingrese el Nombre: Juana
Ingrese el Apellido:
Ingrese el Username (ej: juanperez): juanaap
Ingrese el Email (ej: juan@correo.com): juanaap@gmail.com
Ingrese la Contraseña para este usuario: prueba9874
ADVERTENCIA: Transacción cerrada sin commit. Haciendo rollback...

(Presione Enter para continuar...)
X ¡ERROR! Error al crear usuario y credencial: Error (Trigger INSERT): El apellido solo puede contener letras y espacios.
```

```
TPI_UsuarioCredencial (run-single) #2 × TPI_UsuarioCredencial (run-single) #3 ×
ADVERTENCIA: Transacción cerrada sin commit. Haciendo rollback...

(Presione Enter para continuar...)
✗ ;ERROR! Error al crear usuario y credencial: Error (Trigger INSERT): El apellido solo puede contener letras y espacios.

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Lucas
Ingrese el Apellido: López
Ingrese el Username (ej: juanperez):
Ingrese el Email (ej: juan@correo.com): lucsalopez@contacto.com
Ingrese la Contraseña para este usuario: prueba879658

✗ ;ERROR! El username es obligatorio

(Presione Enter para continuar...)
```

```
Terminal - localhost Git Repository Browser Output - TPI_UsuarioCredencial (run-single) ×
Compiling 1 source file to C:\Users\User\Documents\Gabriela\IUP\Programacion II\Trabajo Práctico Integrador\Trabajo_
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
compile-single:
run-single:
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
=====
SISTEMA DE GESTIÓN DE USUARIOS (v1.0)
=====

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Tomás
Ingrese el Apellido: Cuello
Ingrese el Username (ej: juanperez): tomycuello
Ingrese el Email (ej: juan@correo.com):
Ingrese la Contraseña para este usuario: prueba789

(Presione Enter para continuar...)
✗ ;ERROR! El email es obligatorio y debe tener formato válido (ej: usuario@dominio.com)
```

```
TPI_UsuarioCredencial (run-single) #2 × TPI_UsuarioCredencial (run-single) #3 × TPI_UsuarioCredencial (run-single) #4 ×
compile-single:
run-single:
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
=====
SISTEMA DE GESTIÓN DE USUARIOS (v1.0)
=====

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Andrea
Ingrese el Apellido: Domínguez
Ingrese el Username (ej: juanperez): andredominguez
Ingrese el Email (ej: juan@correo.com): andreadominguez96@outlook.com
Ingrese la Contraseña para este usuario:

(Presione Enter para continuar...)
✗ ;ERROR! El hash de la contraseña es obligatorio y debe tener al menos 8 caracteres
```

CP05 - Validación de Contraseña Corta

Descripción: Verificar longitud mínima de contraseña

```
Terminal - localhost | Git Repository Browser | Output - TPI_UsuarioCredencial (run-single) x
Ingrese el Username (ej: juanperez): tomycuello
Ingrese el Email (ej: juan@correo.com):
Ingrese la Contraseña para este usuario: prueba789

(Presione Enter para continuar...)
X ;ERROR! El email es obligatorio y debe tener formato válido (ej: usuario@dominio.com)

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Hernán
Ingrese el Apellido: Gutierrez
Ingrese el Username (ej: juanperez): hgusuario
Ingrese el Email (ej: juan@correo.com): hg2025@correo.com
Ingrese la Contraseña para este usuario: d123

(Presione Enter para continuar...)
X ;ERROR! El hash de la contraseña es obligatorio y debe tener al menos 8 caracteres
```

CP06 - Rollback por Error en Credencial

Descripción: Verificar rollback ante el fallo de una operación. En este escenario, se ingresa un nombre que excede la cantidad de caracteres permitida.

```
Terminal - localhost  Git Repository Browser  Output - TPI_UsuarioCredencial (run-single) x

Ingrese el Email (ej: juan@correo.com): hg2025@correo.com
Ingrese la Contraseña para este usuario: dl23

(Presione Enter para continuar...)
X ¡ERROR! El hash de la contraseña es obligatorio y debe tener al menos 8 caracteres

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Nombre Muy Extenso Maximiliano Anastasio Sebastián de las Mercedes de la Santa Cruz y del Monte
Ingrese el Apellido: Apellido
Ingrese el Username (ej: juanperez): nombreeextenso
Ingrese el Email (ej: juan@correo.com): mail@usuario.com
Ingrese la Contraseña para este usuario: 3j3mpl0pwd
ADVERTENCIA: Transacción cerrada sin commit. Haciendo rollback...

(Presione Enter para continuar...)
X ¡ERROR! Error al crear usuario y credencial: Data truncation: Data too long for column 'nombre' at row 1
|
```

CP07 - Consulta de Usuario por Username

Descripción: Verificar que se puede buscar un usuario existente por su username único

```
Terminal - localhost | Git Repository Browser | Output - TPI_UsuarioCredencial (run-single) x
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
compile-single:
run-single:
Picked up JAVA_TOOL_OPTIONS: -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
=====
SISTEMA DE GESTIÓN DE USUARIOS (v1.0)
=====

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 3
Ingrese el Username del usuario que desea buscar: pruebausuario

=====
USUARIO ENCONTRADO
=====
-----
ID:          21
Nombre:      prueba usuario
Username:    pruebausuario
Email:       prueba@mail.com
Activo:      Sí
Credencial:  ¡Cargada!

(Presione Enter para continuar...)
```

CP08 - Búsqueda de Usuario Inexistente

Descripción: Verificar el comportamiento del sistema al buscar un usuario que no existe


```
Terminal - localhost | Git Repository Browser | Output - TPI_UsuarioCredencial (run-single) x
>>
>> Por favor, ingrese una opción: 3
Ingrese el Username del usuario que desea buscar: pruebausuario

=====
USUARIO ENCONTRADO
=====
-----
ID:          21
Nombre:      prueba usuario
Username:    pruebausuario
Email:       prueba@mail.com
Activo:      Sí
Credencial:  ¡Cargada!

(Presione Enter para continuar...)

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 3
Ingrese el Username del usuario que desea buscar: noexiste001

⚠ AVISO: No se encontró ningún usuario con el username 'noexiste001'.
(Presione Enter para continuar...)
```

CP09 - Listado de Todos los Usuarios

Descripción: Verificar que se pueden listar todos los usuarios activos del sistema

--- MENÚ PRINCIPAL ---

1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)

0. Salir

Por favor, ingrese una opción: 2

--- LISTA DE USUARIOS ACTIVOS ---

--- LISTA DE USUARIOS ACTIVOS ---

ID: 1
Nombre: Juan Pérez
Username: jperez
Email: juan.perez@example.com
Activo: Si
Credencial: ¡Cargada!

ID: 2
Nombre: María González
Username: mgonzalez
Email: maria.gonzalez@example.com
Activo: Si
Credencial: ¡Cargada!

ID: 3
Nombre: Carlos Rodríguez
Username: crodriguez
Email: carlos.r@example.com
Activo: Si
Credencial: ¡Cargada!

ID: 4
Nombre: Ana Martínez
Username: amartinez
Email: ana.martinez@example.com
Activo: Si
Credencial: ¡Cargada!

ID: 5

Terminal - localhostGit Repository BrowserOutput - TPI_UsuarioCredencial (run-single) ×

ID:5Nombre:Luis SánchezUsername:lsanchezEmail:luis.s@example.comActivo:SíCredencial:¡Cargada!

ID:6Nombre:Sofía GómezUsername:sgomezEmail:sofia.gomez@example.comActivo:SíCredencial:¡Cargada!

ID:7Nombre:Javier DíazUsername:jdiazEmail:javier.diaz@example.comActivo:SíCredencial:¡Cargada!

ID:8Nombre:Laura HernándezUsername:lhernandezEmail:laura.h@example.comActivo:SíCredencial:¡Cargada!

ID:9Nombre:Miguel ÁlvarezUsername:malvarezEmail:miguel.alvarez@example.comActivo:SíCredencial:¡Cargada!

ID:10Nombre:Elena MuñozUsername:emunozEmail:elena.munoz@example.comActivo:SíCredencial:¡Cargada!

ID:11Nombre:Pedro JiménezUsername:pjimenezEmail:pedro.j@example.com

Terminal - localhostGit Repository BrowserOutput - TPI_UsuarioCredencial (run-single) ×

▶▶

▶▶

■

⚙️

ID:12

Nombre:Carmen Ruiz

Username:cruiz

Email:carmen.ruiz@example.com

Activo:Sí

Credencial:¡Cargada!

ID:13

Nombre:Diego Alonso

Username:dalonso

Email:diego.alonso@example.com

Activo:Sí

Credencial:¡Cargada!

ID:14

Nombre:Lucía García

Username:lgarcia

Email:lucia.garcia@example.com

Activo:Sí

Credencial:¡Cargada!

ID:15

Nombre:Adrián López

Username:alopez

Email:adrian.lopez@example.com

Activo:Sí

Credencial:¡Cargada!

ID:16

Nombre:Paula Moreno

Username:pmoreno

Email:paula.m@example.com

Activo:Sí

Credencial:¡Cargada!

ID:17

Nombre:Manuel Vázquez

Username:mvazquez

Email:manuel.vazquez@example.com

Activo:Sí

Credencial:¡Cargada!

ID:18

Nombre:Isabel Domínguez

Username:idinguez

```
Terminal - localhost | Git Repository Browser | Output - TPI_UsuarioCredencial (run-single) x
Credencial: ¡Cargada!
-----
ID: 16
Nombre: Paula Moreno
Username: pmoreno
Email: paula.m@example.com
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 17
Nombre: Manuel Vázquez
Username: mvazquez
Email: manuel.vazquez@example.com
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 18
Nombre: Isabel Domínguez
Username: idominguez
Email: isabel.d@example.com
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 19
Nombre: David Ortega
Username: dortega
Email: david.ortega@example.com
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 20
Nombre: Sara Núñez
Username: snunez
Email: sara.nunez@example.com
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 21
Nombre: prueba usuario
Username: pruebausuario
Email: prueba@mail.com
Activo: Sí
Credencial: ¡Cargada!

(Presione Enter para continuar...)
```

CP10 - Actualización Exitosa de Usuario

Descripción: Verificar que se pueden modificar los datos de un usuario existente

```
Terminal - localhost | Git Repository Browser | Output - TPI_UsuarioCredencial (run-single) x
ID: 20
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 20
Nombre: Sara Núñez
Username: snunez
Email: sara.nunez@example.com
Activo: Sí
Credencial: ¡Cargada!
-----
ID: 21
Nombre: prueba usuario
Username: pruebausuario
Email: prueba@mail.com
Activo: Sí
Credencial: ¡Cargada!
-----
(Presione Enter para continuar...)

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 4
Ingrese el ID del usuario que desea actualizar: 21
-----
ID: 21
Nombre: prueba usuario
Username: pruebausuario
Email: prueba@mail.com
Activo: Sí
Credencial: ¡Cargada!
-----
Ingrese los nuevos datos (deje en blanco para no cambiar):
Ingrese el Nombre: nuevonombre
Ingrese el Apellido: nuevoapellido
Ingrese el Email (ej: juan@correo.com):

✓ ¡ÉXITO! Usuario ID 21 actualizado correctamente.
```

CP11 - Eliminación Lógica de Usuario

Descripción: Verificar que la eliminación marca como eliminado pero no borra físicamente

```
Terminal - localhost | Git Repository Browser | Output - TPI_UsuarioCredencial (run-single) x
>>>
Por favor, ingrese una opción: 4
Ingrese el ID del usuario que desea actualizar: 21
-----
ID:          21
Nombre:      prueba usuario
Username:    pruebausuario
Email:       prueba@mail.com
Activo:      Sí
Credencial:  ;Cargada!

Ingrese los nuevos datos (deje en blanco para no cambiar):
Ingrese el Nombre: nuevonombre
Ingrese el Apellido: nuevoapellido
Ingrese el Email (ej: juan@correo.com):

✓ ¡ÉXITO! Usuario ID 21 actualizado correctamente.

(Presione Enter para continuar...)

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 5
Ingrese el ID del usuario que desea eliminar: 21 ←

⚠ AVISO: ADVERTENCIA: Está a punto de eliminar (baja lógica) a:
-----
ID:          21
Nombre:      nuevonombre nuevoapellido
Username:    pruebausuario
Email:       prueba@mail.com
Activo:      Sí
Credencial:  ;Cargada!
¿Está seguro? (Escriba 'si' para confirmar): si ←

✓ ¡ÉXITO! Usuario pruebausuario (ID 21) eliminado correctamente.

(Presione Enter para continuar...)
```

CP12 - Unicidad de Username

Descripción: Verificar que el username debe ser único en todo el sistema

```
Terminal - localhost | Git Repository Browser | Output X
TPI_UsuarioCredencial (run-single) X | TPI_UsuarioCredencial (run-single) #2 X
(Presione Enter para continuar...)

--- MENÚ PRINCIPAL ---
1. Crear nuevo Usuario (con Credencial)
2. Listar todos los Usuarios
3. Buscar Usuario por Username
4. Actualizar datos de Usuario
5. Eliminar Usuario (Baja lógica)
-----
0. Salir

Por favor, ingrese una opción: 1

=====
CREAR NUEVO USUARIO
=====
Ingrese el Nombre: Isabel
Ingrese el Apellido: Dominguez Nuevo
Ingrese el Username (ej: juanperez): idominguez
Ingrese el Email (ej: juan@correo.com): isadom@mail.com
Ingrese la Contraseña para este usuario: ejemplo!123

(Presione Enter para continuar...)
X ;ERROR! Error al verificar disponibilidad del username: El username 'idominguez' ya está en uso
```

Funcionalidades verificadas:

- Creación de usuarios con credenciales
- Validaciones de integridad de datos
- Transacciones atómicas (commit/rollback)
- Búsquedas y consultas
- Actualizaciones y eliminaciones
- Reglas de negocio 1→1 y unicidad
- Manejo robusto de casos límite

Todos los casos de prueba han sido ejecutados exitosamente demostrando el correcto funcionamiento del sistema según los requerimientos especificados.