

Documentacion deCodigo Java

[Informacion General](#) [Diagrama de Clases](#) [Diagrama de Secuencia](#) [Diagrama de Actividad](#) [Estructura delCodigo](#) [Reporte de Cobertura](#)

Informacion General del Proyecto

Nombre del Proyecto: UserManagementSystem2
Descripcion: Documentacion generada para el archivo input/UserManagementSystem2.java

Diagrama de Clases

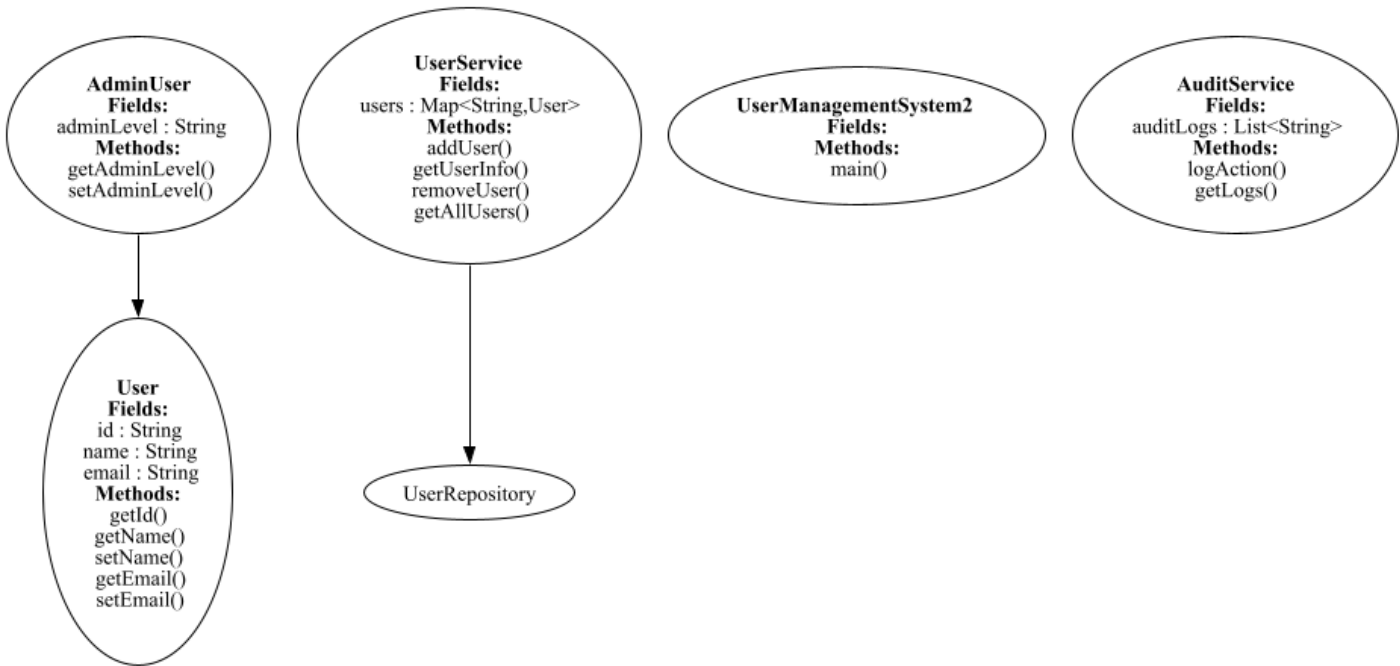


Diagrama de Secuencia

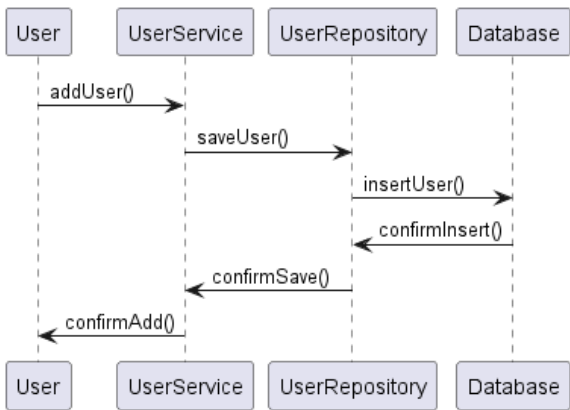
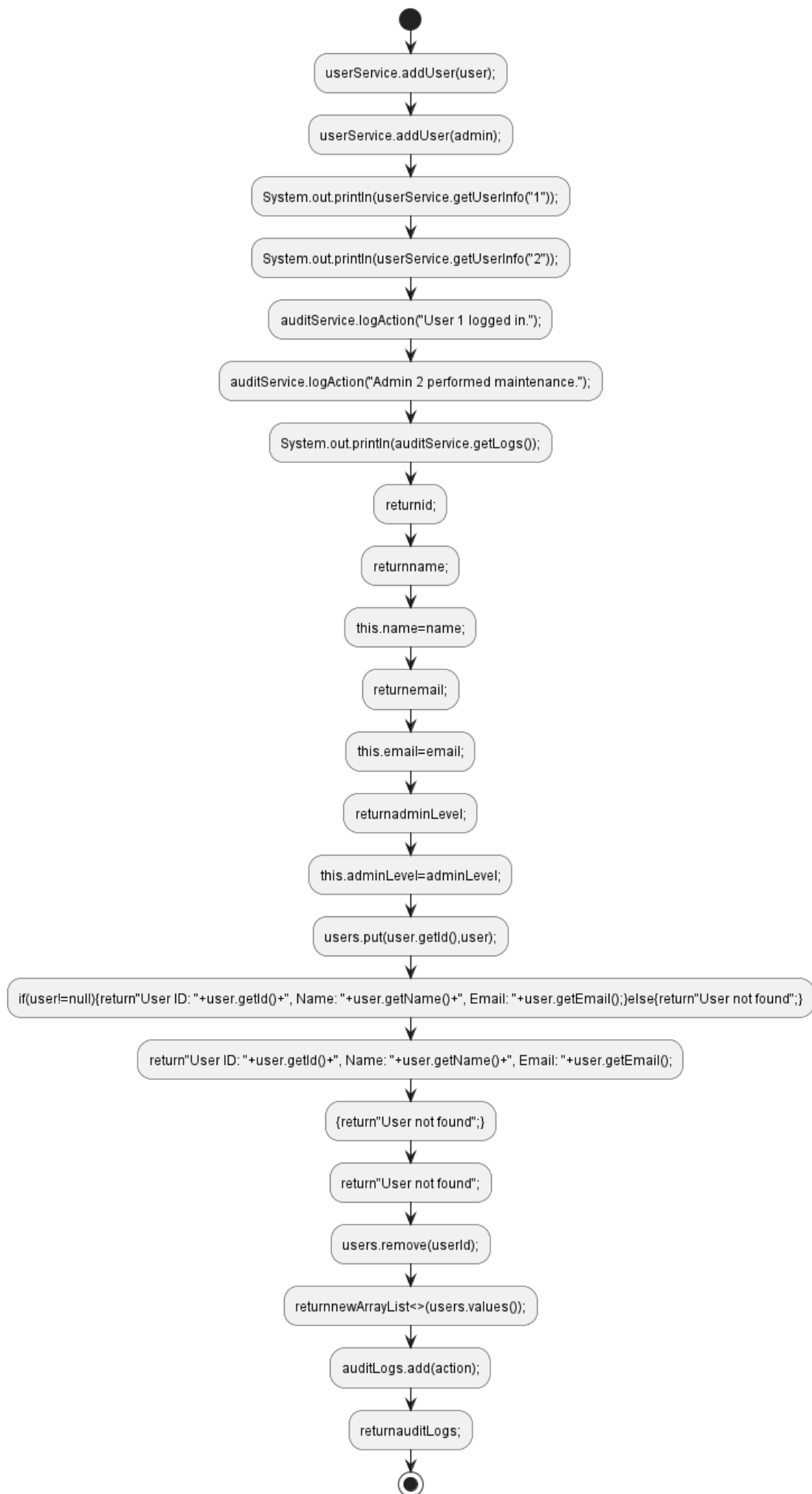


Diagrama de Actividad



Estructura del Codigo

Clase: UserManagementSystem2

Descripcion:

```
/**
 * Clase principal que contiene el punto de entrada al sistema de gesti3n de usuarios.
 */
```

Dependencias:

Metodos

Nombre: main

Tipo de Retorno: void

Descripcion: No hay descripcion.

Modificadores: public static

Parametros:

Descripcion de Parametros:

Clase: User

Descripcion:

```
/**
 * Clase que representa a un usuario en el sistema.
 */
```

Dependencias:

Campos

Nombre: id

Tipo: String

Descripcion:

```
/**
 * Identificador 3nico del usuario.
 */
```

Modificadores: private

Nombre: name

Tipo: String

Descripcion:

```
/**
 * Nombre del usuario.
 */
```

Modificadores: private

Nombre: email

Tipo: String

Descripcion:

```
/**
 * Direcci3n de correo electr3nico del usuario.
 */
```

Modificadores: private

Metodos

Nombre: getld

Tipo de Retorno: String

Descripcion:

```
/**
 * Obtiene el identificador 3nico del usuario.
 * @return El identificador 3nico del usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Nombre: getName

Tipo de Retorno: String

Descripcion:

```
/**
 * Obtiene el nombre del usuario.
 * @return El nombre del usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Nombre: setName

Tipo de Retorno: void

Descripcion:

```
/**
 * Establece el nombre del usuario.
 * @param name El nuevo nombre para el usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Nombre: getEmail

Tipo de Retorno: String

Descripcion:

```
/**
 * Obtiene la direcci3n de correo electr3nico del usuario.
 * @return La direcci3n de correo electr3nico del usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Nombre: setEmail

Tipo de Retorno: void

Descripcion:

```
/**
 * Establece la direcci3n de correo electr3nico del usuario.
 * @param email La nueva direcci3n de correo electr3nico del usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Clase: AdminUser

Descripcion:

```
/**
 * Clase que representa a un usuario administrador, que extiende a la clase User.
 */
```

Dependencias: User

Campos

Nombre: adminLevel

Tipo: String

Descripcion:

```
/**
 * Nivel de administraci3n del usuario.
 */
```

Modificadores: private

Metodos

Nombre: getAdminLevel

Tipo de Retorno: String

Descripcion:

```
/**
 * Obtiene el nivel de administraci3n del usuario.
 * @return El nivel de administraci3n del usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Nombre: setAdminLevel

Tipo de Retorno: void

Descripcion:

```
/**
 * Establece el nivel de administraci3n del usuario.
 * @param adminLevel El nuevo nivel de administraci3n para el usuario.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Clase: UserService

Descripcion:

```
/**
 * Clase de servicio que gestiona las operaciones con los usuarios.
 */
```

Dependencias: UserRepository

Campos

Nombre: users

Tipo: Map

Descripcion:

```
/**
 * Colecci3n de usuarios gestionada por el servicio.
 */
```

Modificadores: private

Metodos

Nombre: addUser

Tipo de Retorno: void

Descripcion:

```
/**
 * Añade un nuevo usuario al repositorio.
 * @param user El usuario a añadir.
 */
```

Modificadores: @Override public

Parametros:

Descripcion de Parametros:

Nombre: getUserInfo

Tipo de Retorno: String

Descripcion:

```
/**
 * Obtiene la información de un usuario basado en su identificador.
 * @param userId El identificador del usuario.
 * @return La información del usuario.
 */
```

Modificadores: @Override public

Parametros:

Descripcion de Parametros:

Nombre: removeUser

Tipo de Retorno: void

Descripcion:

```
/**
 * Elimina un usuario del repositorio.
 * @param userId El identificador del usuario a eliminar.
 */
```

Modificadores: @Override public

Parametros:

Descripcion de Parametros:

Nombre: getAllUsers

Tipo de Retorno: List

Descripcion:

```
/**
 * Obtiene la lista de todos los usuarios.
 * @return La lista de todos los usuarios.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Clase: AuditService

Descripcion:

```
/**
 * Clase de servicio que gestiona el registro de auditorías.
 */
```

Dependencias:

Campos

Nombre: auditLogs

Tipo: List

Descripcion:

```
/**
 * Lista de registros de auditorías.
 */
```

Modificadores: private

Metodos

Nombre: logAction

Tipo de Retorno: void

Descripcion:

```
/**
 * Registra una acción en el sistema.
 * @param action La acción a registrar.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Nombre: getLogs

Tipo de Retorno: List

Descripcion:

```
/**
 * Obtiene todos los registros de auditorías.
 * @return La lista de registros de auditorías.
 */
```

Modificadores: public

Parametros:

Descripcion de Parametros:

Comentarios y Anotaciones

Ã±adir comentarios aquÃ±...

Reporte de Cobertura

JaCoCo Coverage Report

[Sessions](#)

JaCoCo Coverage Report

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
com.example.usermanagement	<div><div></div></div>	79%	<div><div></div></div>	50%	8	20	12	47	7	19
Total	33 of 162	79%	1 of 2	50%	8	20	12	47	7	19

Created with [JaCoCo](#) 0.8.12.202403310830