



# CHAPTER 17

## ユーザ管理関数

この章では、次のユーザ管理関数に関する情報を示します。

- 「createUser」 (P.17-1)
- 「deleteUser」 (P.17-2)
- 「getDiscoveryAuthInfo」 (P.17-3)
- 「listAllUsers」 (P.17-3)
- 「readUserInfo」 (P.17-4)
- 「writeUserInfo」 (P.17-4)

### createUser

#### 構文

```
UserInfo createUser(UserToken token, String username, String password, Role role) throws  
RemoteException;
```

#### 説明

この関数は、新しいユーザを作成します。この関数を呼び出す権限を得るには、管理者ロールを使用してログインする必要があります。ロールが指定されていない場合は、デフォルト ロールの REPORTMGR が割り当てられます。

#### 入力パラメータ

パラメータ	タイプ	値	説明
token	UserToken、 必須	—	ユーザの認証パスを表すトークン。これは、ユーザが login 関数を呼び出して、バックエンドサーバがそのユーザを認証した後に取得されます。
username	String、必須	x21 ~ x7A の範囲の ASCII 文字を含む最大 64 文字の文字列	ユーザ名。

パラメータ	タイプ	値	説明
password	String、必須	x21 ~ x7A の範囲の ASCII 文字を含む最大 64 文字の文字列	ユーザ パスワード。
role	Role、必須	列挙値	Role は、指定したユーザのための列挙値です。

### 戻り値

この関数は、操作が成功した場合、UserInfo オブジェクトを返します。

### エラーと例外

システム エラーによって操作が完了しなかった場合は、RemoteException がスローされます。

エラーが発生した場合、この関数は null を返します。

## deleteUser

### 構文

```
boolean deleteUser(UserToken token, String username) throws RemoteException;
```

### 説明

この関数は、システムからユーザを削除します。この関数を呼び出す権限を得るには、管理者ロールを使用してログインする必要があります。

### 入力パラメータ

パラメータ	タイプ	値	説明
token	UserToken、必須	—	ユーザの認証パスを表すトークン。これは、ユーザが login 関数を呼び出して、バックエンドサーバがそのユーザを認証した後に取得されます。
username	String、必須	—	ユーザ名。

### 戻り値

この関数は、操作が成功した場合は true を返し、失敗した場合は false を返します。

### エラーと例外

システム エラーによって操作が完了しなかった場合は、RemoteException がスローされます。

エラーが発生した場合、この関数は false を返します。

# getDiscoveryAuthInfo

## 構文

```
DiscoveryAuthInfo getDiscoveryAuthInfo (UserToken token) throws RemoteException;
```

## 説明

この関数は、検出時に管理者ユーザによって設定されたユーザ名、パスワード、およびイネーブルパスワードのコレクションを返します。この関数を呼び出す権限を得るには、管理者ロールを使用してログインする必要があります。

## 入力パラメータ

パラメータ	タイプ	値	説明
token	UserToken、 必須	—	ユーザの認証パスを表すトークン。これは、ユーザが <code>login</code> 関数を呼び出して、バックエンドサーバがそのユーザを認証した後に取得されます。

## 戻り値

この関数は `DiscoveryAuthInfo` オブジェクトを返します。

## エラーと例外

システム エラーによって操作が完了しなかった場合は、`RemoteException` がスローされます。

# listAllUsers

## 構文

```
String[] listAllUsers(UserToken token) throws RemoteException;
```

## 説明

この関数は、システムに登録されているユーザ名のリストを返します。この関数を呼び出す権限を得るには、管理者ロールを使用してログインする必要があります。

## 入力パラメータ

パラメータ	タイプ	値	説明
token	UserToken、 必須	—	ユーザの認証パスを表すトークン。これは、ユーザが <code>login</code> 関数を呼び出して、バックエンドサーバがそのユーザを認証した後に取得されます。

## 戻り値

この関数は、ユーザの名前を表す文字列配列を返します。

**エラーと例外**

システム エラーによって操作が完了しなかった場合は、`RemoteException` がスローされます。

エラーが発生した場合、この関数は `null` を返します。

## readUserInfo

**構文**

```
UserInfo readUserInfo(UserToken token, String username) throws RemoteException;
```

**説明**

この関数は、ユーザ情報を取得します。トークンだけを指定した場合は、そのトークンに関連付けられたユーザ情報が取得されます。トークンとユーザ名の両方を指定した場合は、ユーザ名に関連付けられた情報が取得されます。後者の呼び出しは、管理者ロールを持つユーザだけが実行できます。

**入力パラメータ**

パラメータ	タイプ	値	説明
token	UserToken、 必須	—	ユーザの認証パスを表すトークン。これは、ユーザが <code>login</code> 関数を呼び出して、バックエンドサーバがそのユーザを認証した後に取得されます。
username	String、必須	—	ユーザ名。

**戻り値**

この関数は `UserInfo` オブジェクトを返します。

**エラーと例外**

システム エラーによって操作が完了しなかった場合は、`RemoteException` がスローされます。

エラーが発生した場合、この関数は `null` を返します。

## writeUserInfo

**構文**

```
boolean writeUserInfo(UserToken token, UserInfo user) throws RemoteException;
```

**説明**

この関数は、指定した `UserInfo` オブジェクトをインベントリに書き込みます。入力 `UserInfo` オブジェクトは、すでに存在していて、`readUserInfo()` 関数によってインベントリから取得されたものである必要があります。

## 入力パラメータ

パラメータ	タイプ	値	説明
token	UserToken、 必須	—	ユーザの認証パスを表すトークン。これは、ユーザが <b>login</b> 関数を呼び出して、バックエンドサーバがそのユーザを認証した後に取得されます。
user	UserInfo オブジェクト、 必須	—	UserInfo オブジェクト。

## 戻り値

この関数は、操作が成功した場合は **true** を返し、失敗した場合は **false** を返します。

## エラーと例外

システム エラーによって操作が完了しなかった場合は、**RemoteException** がスローされます。

エラーが発生した場合、この関数は **false** を返します。

