

ATM

Generated by Doxygen 1.10.0



<b>1 jQuery v3.6.1</b>	<b>1</b>
1.0.1 jQuery License . . . . .	1
<b>2 jQuery UI v1.13.2</b>	<b>3</b>
2.0.1 jQuery UI License . . . . .	3
<b>3 Hierarchical Index</b>	<b>5</b>
3.1 Class Hierarchy . . . . .	5
<b>4 Class Index</b>	<b>7</b>
4.1 Class List . . . . .	7
<b>5 Class Documentation</b>	<b>9</b>
5.1 com.xiaojun.model.AdminAccount Class Reference . . . . .	9
5.1.1 Detailed Description . . . . .	9
5.2 com.xiaojun.repository.AdminAccountRepository Interface Reference . . . . .	9
5.2.1 Detailed Description . . . . .	9
5.3 com.xiaojun.admin.AdminLogin Class Reference . . . . .	10
5.3.1 Detailed Description . . . . .	10
5.3.2 Constructor & Destructor Documentation . . . . .	10
5.3.2.1 AdminLogin() . . . . .	10
5.3.3 Member Function Documentation . . . . .	10
5.3.3.1 accountLogin() . . . . .	10
5.4 com.xiaojun.admin.AdminLoginTest Class Reference . . . . .	11
5.5 com.xiaojun.admin.AdminUI Class Reference . . . . .	11
5.5.1 Detailed Description . . . . .	11
5.5.2 Constructor & Destructor Documentation . . . . .	11
5.5.2.1 AdminUI() . . . . .	11
5.5.3 Member Function Documentation . . . . .	12
5.5.3.1 showAdminUI() . . . . .	12
5.6 com.xiaojun.admin.AdminUITest Class Reference . . . . .	12
5.7 com.xiaojun.login.ATM Class Reference . . . . .	12
5.7.1 Detailed Description . . . . .	13
5.8 com.xiaojun.login.ATMTest Class Reference . . . . .	13
5.9 com.xiaojun.admin.CreateAccount Class Reference . . . . .	13
5.9.1 Detailed Description . . . . .	13
5.9.2 Constructor & Destructor Documentation . . . . .	13
5.9.2.1 CreateAccount() . . . . .	13
5.9.3 Member Function Documentation . . . . .	14
5.9.3.1 adminOperate() . . . . .	14
5.10 com.xiaojun.admin.CreateAccountTest Class Reference . . . . .	14
5.11 com.xiaojun.model.CustomerAccount Class Reference . . . . .	14
5.11.1 Detailed Description . . . . .	14
5.12 com.xiaojun.repository.CustomerAccountRepository Interface Reference . . . . .	15

5.12.1 Detailed Description	15
5.13 com.xiaojun.customer.CustomerLogin Class Reference	15
5.13.1 Detailed Description	16
5.13.2 Constructor & Destructor Documentation	16
5.13.2.1 CustomerLogin()	16
5.13.3 Member Function Documentation	16
5.13.3.1 accountLogin()	16
5.14 com.xiaojun.customer.CustomerLoginTest Class Reference	16
5.15 com.xiaojun.customer.CustomerUI Class Reference	17
5.15.1 Detailed Description	17
5.15.2 Constructor & Destructor Documentation	17
5.15.2.1 CustomerUI()	17
5.15.3 Member Function Documentation	17
5.15.3.1 showCustomerUI()	17
5.16 com.xiaojun.customer.CustomerUITest Class Reference	18
5.17 com.xiaojun.admin.DeleteAccount Class Reference	18
5.17.1 Detailed Description	18
5.17.2 Constructor & Destructor Documentation	18
5.17.2.1 DeleteAccount()	18
5.17.3 Member Function Documentation	19
5.17.3.1 adminOperate()	19
5.18 com.xiaojun.admin.DeleteAccountTest Class Reference	19
5.19 com.xiaojun.admin.DisplayAccountDetail Class Reference	19
5.19.1 Detailed Description	19
5.19.2 Member Function Documentation	20
5.19.2.1 display()	20
5.20 com.xiaojun.admin.DisplayAccountDetailTest Class Reference	20
5.21 com.xiaojun.customer.DisplayBalance Class Reference	20
5.21.1 Detailed Description	20
5.21.2 Constructor & Destructor Documentation	20
5.21.2.1 DisplayBalance()	20
5.21.3 Member Function Documentation	21
5.21.3.1 customerOperate()	21
5.22 com.xiaojun.customer.DisplayBalanceTest Class Reference	21
5.23 com.xiaojun.admin.IAdminOperate Interface Reference	21
5.23.1 Detailed Description	22
5.23.2 Member Function Documentation	22
5.23.2.1 adminOperate()	22
5.24 com.xiaojun.customer.ICustomerOperate Interface Reference	22
5.24.1 Detailed Description	22
5.24.2 Member Function Documentation	23
5.24.2.1 customerOperate()	23

5.25 com.xiaojun.login.ILogin Interface Reference . . . . .	23
5.25.1 Detailed Description . . . . .	23
5.25.2 Member Function Documentation . . . . .	23
5.25.2.1 accountLogin() . . . . .	23
5.26 com.xiaojun.Main Class Reference . . . . .	24
5.26.1 Detailed Description . . . . .	24
5.26.2 Member Function Documentation . . . . .	24
5.26.2.1 commandLineRunner() . . . . .	24
5.26.2.2 scanner() . . . . .	24
5.27 com.xiaojun.MainTest Class Reference . . . . .	25
5.28 com.xiaojun.admin.SearchAccount Class Reference . . . . .	25
5.28.1 Detailed Description . . . . .	25
5.28.2 Member Function Documentation . . . . .	25
5.28.2.1 adminOperate() . . . . .	25
5.29 com.xiaojun.admin.SearchAccountTest Class Reference . . . . .	25
5.30 com.xiaojun.admin.UpdateAccount Class Reference . . . . .	26
5.30.1 Detailed Description . . . . .	26
5.30.2 Member Function Documentation . . . . .	26
5.30.2.1 adminOperate() . . . . .	26
5.31 com.xiaojun.admin.UpdateAccountTest Class Reference . . . . .	26
5.32 com.xiaojun.customer.UpdateDeposit Class Reference . . . . .	27
5.32.1 Detailed Description . . . . .	27
5.32.2 Constructor & Destructor Documentation . . . . .	27
5.32.2.1 UpdateDeposit() . . . . .	27
5.32.3 Member Function Documentation . . . . .	27
5.32.3.1 customerOperate() . . . . .	27
5.33 com.xiaojun.customer.UpdateDepositTest Class Reference . . . . .	28
5.34 com.xiaojun.customer.Withdraw Class Reference . . . . .	28
5.34.1 Detailed Description . . . . .	28
5.34.2 Constructor & Destructor Documentation . . . . .	28
5.34.2.1 Withdraw() . . . . .	28
5.34.3 Member Function Documentation . . . . .	29
5.34.3.1 customerOperate() . . . . .	29
5.35 com.xiaojun.customer.WithdrawTest Class Reference . . . . .	29

## Index

31



# Chapter 1

## jQuery v3.6.1

### 1.0.1 jQuery License

jQuery v 3.6.1  
Copyright OpenJS Foundation and other contributors, <https://openjsf.org/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

\*\*\*\*\*

The jQuery JavaScript Library v3.6.1 also includes Sizzle.js

Sizzle.js includes the following license:

Copyright JS Foundation and other contributors, <https://js.foundation/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/jquery/sizzle>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION

WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

All files located in the node\_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

\*\*\*\*\*



## Chapter 2

# jQuery UI v1.13.2

### 2.0.1 jQuery UI License

Copyright jQuery Foundation and other contributors, <https://jquery.org/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/jquery/jquery-ui>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code contained within the demos directory.

CC0: <http://creativecommons.org/publicdomain/zero/1.0/>

====

All files located in the node\_modules and external directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.



## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

com.xiaojun.model.AdminAccount . . . . .	9
com.xiaojun.admin.AdminLoginTest . . . . .	11
com.xiaojun.admin.AdminUI . . . . .	11
com.xiaojun.admin.AdminUITest . . . . .	12
com.xiaojun.login.ATM . . . . .	12
com.xiaojun.login.ATMTest . . . . .	13
com.xiaojun.admin.CreateAccountTest . . . . .	14
com.xiaojun.model.CustomerAccount . . . . .	14
com.xiaojun.customer.CustomerLoginTest . . . . .	16
com.xiaojun.customer.CustomerUI . . . . .	17
com.xiaojun.customer.CustomerUITest . . . . .	18
com.xiaojun.admin.DeleteAccountTest . . . . .	19
com.xiaojun.admin.DisplayAccountDetail . . . . .	19
com.xiaojun.admin.DisplayAccountDetailTest . . . . .	20
com.xiaojun.customer.DisplayBalanceTest . . . . .	21
com.xiaojun.admin.IAdminOperate . . . . .	21
com.xiaojun.admin.CreateAccount . . . . .	13
com.xiaojun.admin.DeleteAccount . . . . .	18
com.xiaojun.admin.SearchAccount . . . . .	25
com.xiaojun.admin.UpdateAccount . . . . .	26
com.xiaojun.customer.ICustomerOperate . . . . .	22
com.xiaojun.customer.DisplayBalance . . . . .	20
com.xiaojun.customer.UpdateDeposit . . . . .	27
com.xiaojun.customer.Withdraw . . . . .	28
com.xiaojun.login.ILogin . . . . .	23
com.xiaojun.admin.AdminLogin . . . . .	10
com.xiaojun.customer.CustomerLogin . . . . .	15
com.xiaojun.Main . . . . .	24
com.xiaojun.MainTest . . . . .	25
com.xiaojun.admin.SearchAccountTest . . . . .	25
com.xiaojun.admin.UpdateAccountTest . . . . .	26
com.xiaojun.customer.UpdateDepositTest . . . . .	28
com.xiaojun.customer.WithdrawTest . . . . .	29
JpaRepository	
com.xiaojun.repository.AdminAccountRepository . . . . .	9
com.xiaojun.repository.CustomerAccountRepository . . . . .	15



## Chapter 4

# Class Index

### 4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">com.xiaojun.model.AdminAccount</a>	9
<a href="#">com.xiaojun.repository.AdminAccountRepository</a>	9
<a href="#">com.xiaojun.admin.AdminLogin</a>	10
<a href="#">com.xiaojun.admin.AdminLoginTest</a>	11
<a href="#">com.xiaojun.admin.AdminUI</a>	11
<a href="#">com.xiaojun.admin.AdminUITest</a>	12
<a href="#">com.xiaojun.login.ATM</a>	12
<a href="#">com.xiaojun.login.ATMTest</a>	13
<a href="#">com.xiaojun.admin.CreateAccount</a>	13
<a href="#">com.xiaojun.admin.CreateAccountTest</a>	14
<a href="#">com.xiaojun.model.CustomerAccount</a>	14
<a href="#">com.xiaojun.repository.CustomerAccountRepository</a>	15
<a href="#">com.xiaojun.customer.CustomerLogin</a>	15
<a href="#">com.xiaojun.customer.CustomerLoginTest</a>	16
<a href="#">com.xiaojun.customer.CustomerUI</a>	17
<a href="#">com.xiaojun.customer.CustomerUITest</a>	18
<a href="#">com.xiaojun.admin.DeleteAccount</a>	18
<a href="#">com.xiaojun.admin.DeleteAccountTest</a>	19
<a href="#">com.xiaojun.admin.DisplayAccountDetail</a>	19
<a href="#">com.xiaojun.admin.DisplayAccountDetailTest</a>	20
<a href="#">com.xiaojun.customer.DisplayBalance</a>	20
<a href="#">com.xiaojun.customer.DisplayBalanceTest</a>	21
<a href="#">com.xiaojun.admin.IAdminOperate</a>	21
<a href="#">com.xiaojun.customer.ICustomerOperate</a>	22
<a href="#">com.xiaojun.login.ILogin</a>	23
<a href="#">com.xiaojun.Main</a>	24
<a href="#">com.xiaojun.MainTest</a>	25
<a href="#">com.xiaojun.admin.SearchAccount</a>	25
<a href="#">com.xiaojun.admin.SearchAccountTest</a>	25
<a href="#">com.xiaojun.admin.UpdateAccount</a>	26
<a href="#">com.xiaojun.admin.UpdateAccountTest</a>	26
<a href="#">com.xiaojun.customer.UpdateDeposit</a>	27
<a href="#">com.xiaojun.customer.UpdateDepositTest</a>	28
<a href="#">com.xiaojun.customer.Withdraw</a>	28
<a href="#">com.xiaojun.customer.WithdrawTest</a>	29



## Chapter 5

# Class Documentation

### 5.1 com.xiaojun.model.AdminAccount Class Reference

#### 5.1.1 Detailed Description

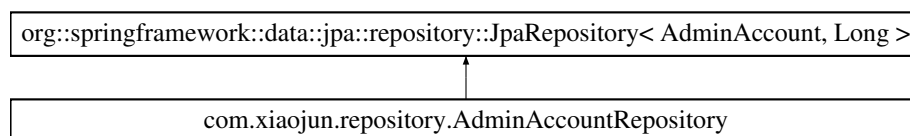
Represents an administrator account in the system. This entity is mapped to a database table with fields for account identification, login credentials, and security pin code.

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/model/AdminAccount.java

### 5.2 com.xiaojun.repository.AdminAccountRepository Interface Reference

Inheritance diagram for com.xiaojun.repository.AdminAccountRepository:



#### Public Member Functions

- [AdminAccount](#) **findByLoginAndPincode** (String loginname, String pincode)

#### 5.2.1 Detailed Description

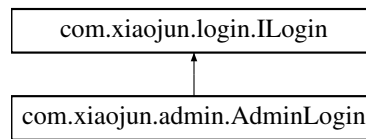
Repository interface for `AdminAccount` data operations. Extends `JpaRepository` to provide basic CRUD operations and query method execution for admin accounts. Utilizes Spring Data JPA to reduce boilerplate data access code and improve implementation efficiency.

The documentation for this interface was generated from the following file:

- atm/src/main/java/com/xiaojun/repository/AdminAccountRepository.java

## 5.3 com.xiaojun.admin.AdminLogin Class Reference

Inheritance diagram for com.xiaojun.admin.AdminLogin:



### Public Member Functions

- [AdminLogin](#) (Scanner scanner, [AdminUI](#) adminUI, [AdminAccountRepository](#) adminAccountRepository)
- void [accountLogin](#) ()

### 5.3.1 Detailed Description

Component responsible for handling the login process for administrators. It facilitates authentication against stored admin credentials and provides access to the admin user interface upon successful login.

### 5.3.2 Constructor & Destructor Documentation

#### 5.3.2.1 AdminLogin()

```

com.xiaojun.admin.AdminLogin.AdminLogin (
    Scanner scanner,
    AdminUI adminUI,
    AdminAccountRepository adminAccountRepository )
  
```

Constructs an AdminLogin instance with necessary dependencies.

#### Parameters

<i>scanner</i>	Scanner object to handle user input.
<i>adminUI</i>	User interface component for admin functions.
<i>adminAccountRepository</i>	Repository to access admin account credentials.

### 5.3.3 Member Function Documentation

#### 5.3.3.1 accountLogin()

```

void com.xiaojun.admin.AdminLogin.accountLogin ( )
  
```

Handles the account login process for administrators. Prompts for username and password, validates them against the repository, and grants access to the admin UI upon successful authentication. Users can exit the login process by typing 'exit'.



Implements [com.xiaojun.login.ILogin](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/AdminLogin.java

## 5.4 com.xiaojun.admin.AdminLoginTest Class Reference

### Public Member Functions

- void **setup** ()
- void **restoreStreams** ()
- void **testExitLoginProcess** ()
- void **testSuccessfulLogin** ()
- void **testFailedLogin** ()
- void **testDataAccessException** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/AdminLoginTest.java

## 5.5 com.xiaojun.admin.AdminUI Class Reference

### Public Member Functions

- [AdminUI](#) (Scanner scanner, [IAdminOperate](#) createAccount, [IAdminOperate](#) deleteAccount, [IAdminOperate](#) updateAccount, [IAdminOperate](#) searchAccount)
- void [showAdminUI](#) ()

### 5.5.1 Detailed Description

Component that serves as the user interface for administrative actions within the application. This class handles user interactions and delegates administrative tasks such as creating, deleting, updating, and searching for accounts to specific components that implement the [IAdminOperate](#) interface.

### 5.5.2 Constructor & Destructor Documentation

#### 5.5.2.1 AdminUI()

```
com.xiaojun.admin.AdminUI.AdminUI (
    Scanner scanner,
    IAdminOperate createAccount,
    IAdminOperate deleteAccount,
    IAdminOperate updateAccount,
    IAdminOperate searchAccount )
```

Constructs an AdminUI with the necessary components and a scanner for input.

**Parameters**

<i>scanner</i>	The scanner to read user commands.
<i>createAccount</i>	Component responsible for creating accounts.
<i>deleteAccount</i>	Component responsible for deleting accounts.
<i>updateAccount</i>	Component responsible for updating account information.
<i>searchAccount</i>	Component responsible for searching for accounts.

### 5.5.3 Member Function Documentation

#### 5.5.3.1 showAdminUI()

```
void com.xiaojun.admin.AdminUI.showAdminUI ( )
```

Displays the administrative user interface and handles the selection of various administrative operations. Continuously prompts the user to choose an action and delegates the task to the appropriate component.

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/AdminUI.java

## 5.6 com.xiaojun.admin.AdminUITest Class Reference

**Public Member Functions**

- void **setUp** ()
- void **testCreateAccountOptionSelected** ()
- void **testDeleteAccountOptionSelected** ()
- void **testUpdateAccountOptionSelected** ()
- void **testSearchAccountOptionSelected** ()
- void **testExitOptionSelected** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/AdminUITest.java

## 5.7 com.xiaojun.login.ATM Class Reference

**Public Member Functions**

- void **login** (ILogin ilogin)

### 5.7.1 Detailed Description

The `ATM` class represents an ATM machine simulation in terms of user authentication. It uses dependency injection to utilize different login mechanisms.

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/login/ATM.java

## 5.8 com.xiaojun.login.ATMTest Class Reference

### Public Member Functions

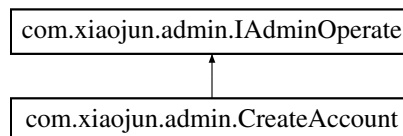
- void **setUp** ()
- void **testLoginSuccess** ()
- void **testLoginFailure** ()
- void **testLoginThrowsException** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/login/ATMTest.java

## 5.9 com.xiaojun.admin.CreateAccount Class Reference

Inheritance diagram for `com.xiaojun.admin.CreateAccount`:



### Public Member Functions

- [CreateAccount](#) (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository)
- void [adminOperate](#) ()

### 5.9.1 Detailed Description

Service class that implements the administrative operation for creating new customer accounts. This class handles the user input necessary to gather all details required to create a new account in the system.

### 5.9.2 Constructor & Destructor Documentation

#### 5.9.2.1 CreateAccount()

```

com.xiaojun.admin.CreateAccount.CreateAccount (
    Scanner scanner,
    CustomerAccountRepository customerAccountRepository )
  
```

Constructor for `CreateAccount` service with dependencies injected via Spring's `@Autowired` annotation.

## Parameters

<i>scanner</i>	The scanner object to handle user input.
<i>customerAccountRepository</i>	The repository used for customer account operations.

### 5.9.3 Member Function Documentation

#### 5.9.3.1 adminOperate()

```
void com.xiaojun.admin.CreateAccount.adminOperate ( )
```

Implements the admin operation to create a new customer account. Prompts the administrator for account details and commits them to the database if they meet validation criteria.

Implements [com.xiaojun.admin.IAdminOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/CreateAccount.java

## 5.10 com.xiaojun.admin.CreateAccountTest Class Reference

### Public Member Functions

- void **setUp** ()
- void **testCreateAccount\_Successful** ()
- void **testCreateAccount\_Failure\_ExistingUsername** ()
- void **testCreateAccount\_InvalidPinCode** ()
- void **testExitCreationProcess** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/CreateAccountTest.java

## 5.11 com.xiaojun.model.CustomerAccount Class Reference

### 5.11.1 Detailed Description

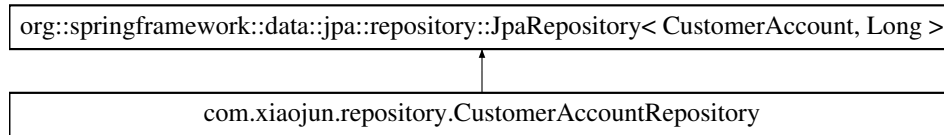
Represents a customer account in a financial or service-oriented system. This entity class is used to map customer account details to a database table including information such as username, pincode, name, balance, and status.

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/model/CustomerAccount.java

## 5.12 com.xiaojun.repository.CustomerAccountRepository Interface Reference

Inheritance diagram for com.xiaojun.repository.CustomerAccountRepository:



### Public Member Functions

- [CustomerAccount](#) **findByUsername** (String loginName)
- [CustomerAccount](#) **findByUsernameAndPincode** (String login, String pinCode)
- [CustomerAccount](#) **findByAccountid** (Long accountId)
- [CustomerAccount](#) **saveAndFlush** ([CustomerAccount](#) customerAccount)

### 5.12.1 Detailed Description

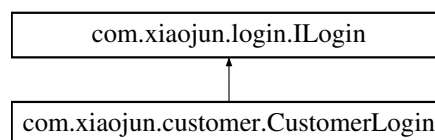
Repository interface for `CustomerAccount` entities. Extends `JpaRepository` to provide standard CRUD operations and query executions for `CustomerAccount` entities. This interface leverages Spring Data JPA to automatically generate implementations that handle the database operations required for managing `CustomerAccount` records, thereby reducing the need for explicit SQL.

The documentation for this interface was generated from the following file:

- atm/src/main/java/com/xiaojun/repository/CustomerAccountRepository.java

## 5.13 com.xiaojun.customer.CustomerLogin Class Reference

Inheritance diagram for com.xiaojun.customer.CustomerLogin:



### Public Member Functions

- [CustomerLogin](#) (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository, [CustomerUI](#) customerUI)
- void [accountLogin](#) ()

### 5.13.1 Detailed Description

Component responsible for managing customer login processes. This class handles the authentication of customer credentials and provides access to the customer user interface upon successful login.

### 5.13.2 Constructor & Destructor Documentation

#### 5.13.2.1 CustomerLogin()

```
com.xiaojun.customer.CustomerLogin.CustomerLogin (
    Scanner scanner,
    CustomerAccountRepository customerAccountRepository,
    CustomerUI customerUI )
```

Constructs a CustomerLogin with necessary dependencies for handling login processes.

##### Parameters

<i>scanner</i>	Scanner object to read user inputs.
<i>customerAccountRepository</i>	Repository to interact with customer account data.
<i>customerUI</i>	User interface component for customer interactions after login.

### 5.13.3 Member Function Documentation

#### 5.13.3.1 accountLogin()

```
void com.xiaojun.customer.CustomerLogin.accountLogin ( )
```

Handles the account login process for customers. Prompts for login and pin code, validates them against the stored credentials, and grants access to the customer UI upon successful authentication. If authentication fails, the user is prompted to try again.

Implements [com.xiaojun.login.ILogin](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/customer/CustomerLogin.java

## 5.14 com.xiaojun.customer.CustomerLoginTest Class Reference

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/customer/CustomerLoginTest.java

## 5.15 com.xiaojun.customer.CustomerUI Class Reference

### Public Member Functions

- [CustomerUI](#) (Scanner scanner, [ICustomerOperate](#) displayBalance, [ICustomerOperate](#) updateDeposit, [ICustomerOperate](#) withdraw)
- void [showCustomerUI](#) (String login)

### 5.15.1 Detailed Description

Component that serves as the user interface for customer actions within the application. This class manages customer interactions by providing options for various account operations like withdrawing, depositing, and displaying balance.

### 5.15.2 Constructor & Destructor Documentation

#### 5.15.2.1 CustomerUI()

```
com.xiaojun.customer.CustomerUI.CustomerUI (
    Scanner scanner,
    ICustomerOperate displayBalance,
    ICustomerOperate updateDeposit,
    ICustomerOperate withdraw )
```

Constructs a CustomerUI with necessary components for handling customer operations.

#### Parameters

<i>scanner</i>	Scanner object to read user commands.
<i>displayBalance</i>	Component that displays the balance of the customer.
<i>updateDeposit</i>	Component that handles deposit operations.
<i>withdraw</i>	Component that handles withdrawal operations.

### 5.15.3 Member Function Documentation

#### 5.15.3.1 showCustomerUI()

```
void com.xiaojun.customer.CustomerUI.showCustomerUI (
    String login )
```

Displays the customer user interface and handles the selection of various banking operations. Continuously prompts the customer to choose an action and delegates the task to the appropriate component based on the selection.

#### Parameters

<i>login</i>	The login identifier of the customer using the interface.
--------------	---

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/customer/CustomUI.java

## 5.16 com.xiaojun.customer.CustomerUITest Class Reference

### Public Member Functions

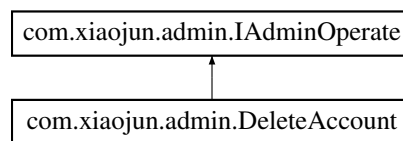
- void **setUp** ()
- void **testWithdrawOptionSelected** ()
- void **testDepositOptionSelected** ()
- void **testDisplayBalanceOptionSelected** ()
- void **testExitOptionSelected** ()
- void **testInvalidInputHandled** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/customer/CustomerUITest.java

## 5.17 com.xiaojun.admin.DeleteAccount Class Reference

Inheritance diagram for com.xiaojun.admin.DeleteAccount:



### Public Member Functions

- [DeleteAccount](#) (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository)
- void [adminOperate](#) ()

### 5.17.1 Detailed Description

Service class that implements the administrative operation for deleting customer accounts. This class handles the user input required to gather details for securely deleting an existing account from the system.

### 5.17.2 Constructor & Destructor Documentation

#### 5.17.2.1 DeleteAccount()

```
com.xiaojun.admin.DeleteAccount.DeleteAccount (
    Scanner scanner,
    CustomerAccountRepository customerAccountRepository )
```

Constructor for DeleteAccount service, initializes the scanner used for input.



## Parameters

<i>scanner</i>	The scanner object to handle user input.
----------------	--

### 5.17.3 Member Function Documentation

#### 5.17.3.1 adminOperate()

```
void com.xiaojun.admin.DeleteAccount.adminOperate ( )
```

Implements the admin operation to delete a customer account. Prompts the administrator to confirm the deletion before permanently removing the account from the database.

Implements [com.xiaojun.admin.IAdminOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/DeleteAccount.java

## 5.18 com.xiaojun.admin.DeleteAccountTest Class Reference

### Public Member Functions

- void **setUp** ()
- void **restoreStreams** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/DeleteAccountTest.java

## 5.19 com.xiaojun.admin.DisplayAccountDetail Class Reference

### Public Member Functions

- void [display](#) (Long id)

#### 5.19.1 Detailed Description

Marks this class as a Spring component, making it eligible for Spring's component scanning to detect and register it as a bean in the Spring application context.

## 5.19.2 Member Function Documentation

### 5.19.2.1 display()

```
void com.xiaojun.admin.DisplayAccountDetail.display (
    Long id )
```

Method to display account details for a given account ID.

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/DisplayAccountDetail.java

## 5.20 com.xiaojun.admin.DisplayAccountDetailTest Class Reference

### Public Member Functions

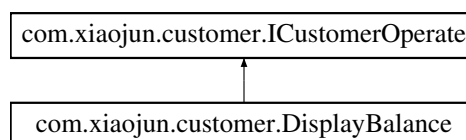
- void **setup** ()
- void **testDisplayAccountExists** ()
- void **testDisplayAccountException** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/DisplayAccountDetailTest.java

## 5.21 com.xiaojun.customer.DisplayBalance Class Reference

Inheritance diagram for com.xiaojun.customer.DisplayBalance:



### Public Member Functions

- [DisplayBalance](#) (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository)
- void [customerOperate](#) (String login)

### 5.21.1 Detailed Description

Service class that implements the customer operation for displaying account balances. This class retrieves the balance information from the database and displays it to the customer.

### 5.21.2 Constructor & Destructor Documentation

#### 5.21.2.1 DisplayBalance()

```
com.xiaojun.customer.DisplayBalance.DisplayBalance (
    Scanner scanner,
    CustomerAccountRepository customerAccountRepository )
```

Constructs a DisplayBalance service with dependencies injected via Spring's @Autowired annotation.

## Parameters

<i>scanner</i>	Scanner object to read user inputs.
<i>customerAccountRepository</i>	Repository to interact with customer account data.

### 5.21.3 Member Function Documentation

#### 5.21.3.1 customerOperate()

```
void com.xiaojun.customer.DisplayBalance.customerOperate (
    String login )
```

Executes the operation to display the balance of a customer's account. Retrieves the account details based on the provided login identifier and displays the current balance along with the date and time of the inquiry.

## Parameters

<i>login</i>	The login identifier of the customer whose balance is being queried.
--------------	--

Implements [com.xiaojun.customer.ICustomerOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/customer/DisplayBalance.java

## 5.22 com.xiaojun.customer.DisplayBalanceTest Class Reference

## Public Member Functions

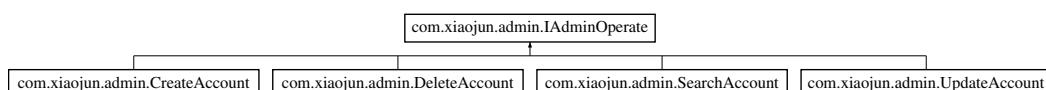
- void **setup** ()
- void **testCustomerOperate\_DisplaySuccessful** ()

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/customer/DisplayBalanceTest.java

## 5.23 com.xiaojun.admin.IAdminOperate Interface Reference

Inheritance diagram for com.xiaojun.admin.IAdminOperate:



## Public Member Functions

- void [adminOperate](#) ()

### 5.23.1 Detailed Description

Interface for defining administrative operations within the application. This interface ensures that classes which implement it provide a specific method for administrative actions.

### 5.23.2 Member Function Documentation

#### 5.23.2.1 adminOperate()

```
void com.xiaojun.admin.IAdminOperate.adminOperate ( )
```

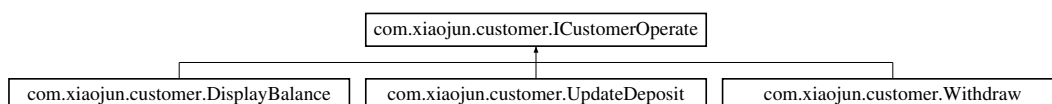
Implemented in [com.xiaojun.admin.CreateAccount](#), [com.xiaojun.admin.DeleteAccount](#), [com.xiaojun.admin.SearchAccount](#), and [com.xiaojun.admin.UpdateAccount](#).

The documentation for this interface was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/IAdminOperate.java

## 5.24 com.xiaojun.customer.ICustomerOperate Interface Reference

Inheritance diagram for com.xiaojun.customer.ICustomerOperate:



## Public Member Functions

- void [customerOperate](#) (String login)

### 5.24.1 Detailed Description

Interface for defining customer operations within the application. This interface ensures that classes which implement it provide specific functionality related to handling customer activities.

## 5.24.2 Member Function Documentation

### 5.24.2.1 customerOperate()

```
void com.xiaojun.customer.ICustomerOperate.customerOperate (
    String login )
```

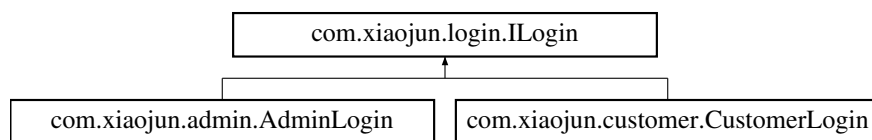
Implemented in [com.xiaojun.customer.DisplayBalance](#), [com.xiaojun.customer.UpdateDeposit](#), and [com.xiaojun.customer.Withdraw](#).

The documentation for this interface was generated from the following file:

- atm/src/main/java/com/xiaojun/customer/ICustomerOperate.java

## 5.25 com.xiaojun.login.ILogin Interface Reference

Inheritance diagram for com.xiaojun.login.ILogin:



### Public Member Functions

- void [accountLogin](#) ()

## 5.25.1 Detailed Description

The `ILogin` interface defines a contract for user login mechanisms. Implementing this interface allows for various authentication strategies to be easily integrated and managed within an application.

## 5.25.2 Member Function Documentation

### 5.25.2.1 accountLogin()

```
void com.xiaojun.login.ILogin.accountLogin ( )
```

Implemented in [com.xiaojun.admin.AdminLogin](#), and [com.xiaojun.customer.CustomerLogin](#).

The documentation for this interface was generated from the following file:

- atm/src/main/java/com/xiaojun/login/ILogin.java

## 5.26 com.xiaojun.Main Class Reference

### Public Member Functions

- CommandLineRunner [commandLineRunner](#) (ApplicationContext ctx)
- Scanner [scanner](#) ()

### Static Public Member Functions

- static void **main** (String[] args)

### 5.26.1 Detailed Description

Main application class that bootstraps a Spring Boot application. This class defines the main entry point of the application and configures Spring context with specific beans.

### 5.26.2 Member Function Documentation

#### 5.26.2.1 `commandLineRunner()`

```
CommandLineRunner com.xiaojun.Main.commandLineRunner (
    ApplicationContext ctx )
```

Bean definition for a CommandLineRunner that includes the user interface for login selection. This runner allows the user to choose between different user types for interaction.

#### Parameters

<code>ctx</code>	Spring's application context, providing access to beans.
------------------	--

#### Returns

a CommandLineRunner with embedded operational logic.

#### 5.26.2.2 `scanner()`

```
Scanner com.xiaojun.Main.scanner ( )
```

Bean definition for creating a Scanner instance. This bean is used across the application to read input from the console.

#### Returns

a new Scanner instance configured to read from the standard input stream.

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/Main.java

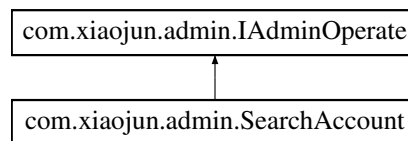
## 5.27 com.xiaojun.MainTest Class Reference

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/MainTest.java

## 5.28 com.xiaojun.admin.SearchAccount Class Reference

Inheritance diagram for com.xiaojun.admin.SearchAccount:



### Public Member Functions

- **SearchAccount** (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository)
- void [adminOperate](#) ()

### 5.28.1 Detailed Description

Service class for searching customer accounts. Implements administrative operations allowing an admin to query and display details of customer accounts.

### 5.28.2 Member Function Documentation

#### 5.28.2.1 adminOperate()

```
void com.xiaojun.admin.SearchAccount.adminOperate ( )
```

Administers the operation for searching customer accounts based on account number. Continuously prompts the admin to enter account numbers to search and displays account details. Allows the admin to exit or continue searching.

Implements [com.xiaojun.admin.IAdminOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/SearchAccount.java

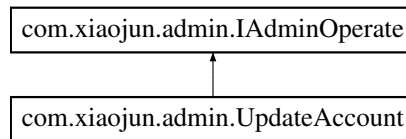
## 5.29 com.xiaojun.admin.SearchAccountTest Class Reference

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/SearchAccountTest.java

## 5.30 com.xiaojun.admin.UpdateAccount Class Reference

Inheritance diagram for com.xiaojun.admin.UpdateAccount:



### Public Member Functions

- **UpdateAccount** (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository, [DisplayAccountDetail](#) displayAccountDetail)
- void [adminOperate](#) ()

### 5.30.1 Detailed Description

Service class for updating customer account details. This class implements administrative operations that allow an admin to update various attributes of a customer's account.

### 5.30.2 Member Function Documentation

#### 5.30.2.1 adminOperate()

```
void com.xiaojun.admin.UpdateAccount.adminOperate ( )
```

Administers the operation for updating customer accounts. Continuously prompts the admin to enter account numbers to update and allows editing of specific account details. The process can be exited at any time by entering 'exit'.

Implements [com.xiaojun.admin.IAdminOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/admin/UpdateAccount.java

## 5.31 com.xiaojun.admin.UpdateAccountTest Class Reference

### Public Member Functions

- void **setup** ()
- void **testAdminOperate\_ReturnAtStart** ()
- void **testAdminOperate\_UpdateUsernameSuccess** ()
- void **testAdminOperate\_InvalidPinCode** ()

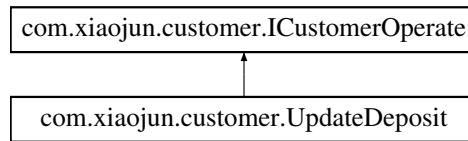
The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/admin/UpdateAccountTest.java



## 5.32 com.xiaojun.customer.UpdateDeposit Class Reference

Inheritance diagram for com.xiaojun.customer.UpdateDeposit:



### Public Member Functions

- [UpdateDeposit](#) (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository)
- void [customerOperate](#) (String login)

### 5.32.1 Detailed Description

Service class that implements the operation for depositing cash into a customer's account. This class manages the input of deposit amounts and updates the account balance accordingly.

### 5.32.2 Constructor & Destructor Documentation

#### 5.32.2.1 UpdateDeposit()

```
com.xiaojun.customer.UpdateDeposit.UpdateDeposit (
    Scanner scanner,
    CustomerAccountRepository customerAccountRepository )
```

Constructs an UpdateDeposit service with dependencies injected via Spring's @Autowired annotation.

#### Parameters

<i>scanner</i>	Scanner object to read user inputs.
<i>customerAccountRepository</i>	Repository to interact with customer account data.

### 5.32.3 Member Function Documentation

#### 5.32.3.1 customerOperate()

```
void com.xiaojun.customer.UpdateDeposit.customerOperate (
    String login )
```

Executes the deposit operation for a customer's account based on the provided login identifier. Prompts the user to enter an amount to deposit, updates the account balance, and confirms the transaction with a detailed receipt.

**Parameters**

<i>login</i>	The login identifier of the customer making the deposit.
--------------	--

Implements [com.xiaojun.customer.ICustomerOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/customer/UpdateDeposit.java

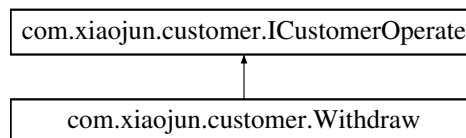
### 5.33 com.xiaojun.customer.UpdateDepositTest Class Reference

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/customer/UpdateDepositTest.java

### 5.34 com.xiaojun.customer.Withdraw Class Reference

Inheritance diagram for com.xiaojun.customer.Withdraw:

**Public Member Functions**

- [Withdraw](#) (Scanner scanner, [CustomerAccountRepository](#) customerAccountRepository)
- void [customerOperate](#) (String login)

#### 5.34.1 Detailed Description

Service class that implements the operation for withdrawing cash from a customer's account. This class manages the withdrawal process, ensuring that the amount withdrawn does not exceed the account balance.

#### 5.34.2 Constructor & Destructor Documentation

##### 5.34.2.1 Withdraw()

```

com.xiaojun.customer.Withdraw.Withdraw (
    Scanner scanner,
    CustomerAccountRepository customerAccountRepository )
  
```

Constructs a Withdraw service with dependencies injected via Spring's @Autowired annotation.

## Parameters

<i>scanner</i>	Scanner object to read user inputs.
<i>customerAccountRepository</i>	Repository to interact with customer account data.

### 5.34.3 Member Function Documentation

#### 5.34.3.1 customerOperate()

```
void com.xiaojun.customer.Withdraw.customerOperate (
    String login )
```

Executes the withdrawal operation for a customer's account based on the provided login identifier. Prompts the user to enter an amount to withdraw, checks if the balance is sufficient, and updates the account accordingly.

## Parameters

<i>login</i>	The login identifier of the customer making the withdrawal.
--------------	---

Implements [com.xiaojun.customer.ICustomerOperate](#).

The documentation for this class was generated from the following file:

- atm/src/main/java/com/xiaojun/customer/Withdraw.java

## 5.35 com.xiaojun.customer.WithdrawTest Class Reference

The documentation for this class was generated from the following file:

- atm/src/test/java/com/xiaojun/customer/WithdrawTest.java



# Index

- accountLogin
  - com.xiaojun.admin.AdminLogin, 10
  - com.xiaojun.customer.CustomerLogin, 16
  - com.xiaojun.login.ILogin, 23
- AdminLogin
  - com.xiaojun.admin.AdminLogin, 10
- adminOperate
  - com.xiaojun.admin.CreateAccount, 14
  - com.xiaojun.admin.DeleteAccount, 19
  - com.xiaojun.admin.IAdminOperate, 22
  - com.xiaojun.admin.SearchAccount, 25
  - com.xiaojun.admin.UpdateAccount, 26
- AdminUI
  - com.xiaojun.admin.AdminUI, 11
- com.xiaojun.admin.AdminLogin, 10
  - accountLogin, 10
  - AdminLogin, 10
- com.xiaojun.admin.AdminLoginTest, 11
- com.xiaojun.admin.AdminUI, 11
  - AdminUI, 11
  - showAdminUI, 12
- com.xiaojun.admin.AdminUITest, 12
- com.xiaojun.admin.CreateAccount, 13
  - adminOperate, 14
  - CreateAccount, 13
- com.xiaojun.admin.CreateAccountTest, 14
- com.xiaojun.admin.DeleteAccount, 18
  - adminOperate, 19
  - DeleteAccount, 18
- com.xiaojun.admin.DeleteAccountTest, 19
- com.xiaojun.admin.DisplayAccountDetail, 19
  - display, 20
- com.xiaojun.admin.DisplayAccountDetailTest, 20
- com.xiaojun.admin.IAdminOperate, 21
  - adminOperate, 22
- com.xiaojun.admin.SearchAccount, 25
  - adminOperate, 25
- com.xiaojun.admin.SearchAccountTest, 25
- com.xiaojun.admin.UpdateAccount, 26
  - adminOperate, 26
- com.xiaojun.admin.UpdateAccountTest, 26
- com.xiaojun.customer.CustomerLogin, 15
  - accountLogin, 16
  - CustomerLogin, 16
- com.xiaojun.customer.CustomerLoginTest, 16
- com.xiaojun.customer.CustomerUI, 17
  - CustomerUI, 17
  - showCustomerUI, 17
- com.xiaojun.customer.CustomerUITest, 18
- com.xiaojun.customer.DisplayBalance, 20
  - customerOperate, 21
  - DisplayBalance, 20
- com.xiaojun.customer.DisplayBalanceTest, 21
- com.xiaojun.customer.ICustomerOperate, 22
  - customerOperate, 23
- com.xiaojun.customer.UpdateDeposit, 27
  - customerOperate, 27
  - UpdateDeposit, 27
- com.xiaojun.customer.UpdateDepositTest, 28
- com.xiaojun.customer.Withdraw, 28
  - customerOperate, 29
  - Withdraw, 28
- com.xiaojun.customer.WithdrawTest, 29
- com.xiaojun.login.ATM, 12
- com.xiaojun.login.ATMTest, 13
- com.xiaojun.login.ILogin, 23
  - accountLogin, 23
- com.xiaojun.Main, 24
  - commandLineRunner, 24
  - scanner, 24
- com.xiaojun.MainTest, 25
- com.xiaojun.model.AdminAccount, 9
- com.xiaojun.model.CustomerAccount, 14
- com.xiaojun.repository.AdminAccountRepository, 9
- com.xiaojun.repository.CustomerAccountRepository, 15
- commandLineRunner
  - com.xiaojun.Main, 24
- CreateAccount
  - com.xiaojun.admin.CreateAccount, 13
- CustomerLogin
  - com.xiaojun.customer.CustomerLogin, 16
- customerOperate
  - com.xiaojun.customer.DisplayBalance, 21
  - com.xiaojun.customer.ICustomerOperate, 23
  - com.xiaojun.customer.UpdateDeposit, 27
  - com.xiaojun.customer.Withdraw, 29
- CustomerUI
  - com.xiaojun.customer.CustomerUI, 17
- DeleteAccount
  - com.xiaojun.admin.DeleteAccount, 18
- display
  - com.xiaojun.admin.DisplayAccountDetail, 20
- DisplayBalance
  - com.xiaojun.customer.DisplayBalance, 20
- jQuery UI v1.13.2, 3
- jQuery v3.6.1, 1

scanner  
    com.xiaojun.Main, [24](#)  
showAdminUI  
    com.xiaojun.admin.AdminUI, [12](#)  
showCustomerUI  
    com.xiaojun.customer.CustomerUI, [17](#)  
  
UpdateDeposit  
    com.xiaojun.customer.UpdateDeposit, [27](#)  
  
Withdraw  
    com.xiaojun.customer.Withdraw, [28](#)