CRC Cards

By: Max, Pierre, Jason, Dian Yu, Iftekhar, Rahdin

Entities

Account

Responsibility

- Store information about an account (username, hashed password, login history, ban state, whether the account is an admin)
- Get an account’s username
- Set an account’s username
- Get an account’s hashed password
- Update login history (by adding a date)
- Get login history
- Check whether an account is an admin
- Promote an account to be an admin
- Set the ban state of the account

Collaborators

Use Cases

AccountManager

Implements: IAccountManager

Responsibility

- Store a mapping of usernames to accounts
- Check if a username is taken by an existing account
- Check whether an account is an admin (based on the account's username)
- Get an account based on a username
- Add a new username-account pair to the mapping
- Delete a username-account pair given a username
- Validate credentials for a log in given a username and password
- Ban / unban an account given its username
- Convert the mapping of usernames to accounts to JSON and vice versa
- Sign up a new user given its username and password
- Create a new admin given its username and password
- Promote an account to admin given its username
- Get an account's login history given its username
- Throw appropriate exceptions when performing the aforementioned responsibilities

Collaborators

- IAccountManager
- Account
- IJsonToObject (implemented by JsonToObject)
- IObjectToJson (implemented by ObjectToJson)
- IHash (implemented by PasswordHash)
- IJsonParser (implemented by JsonParser)
- IncorrectPasswordException
- UsernameExistsException
- UsernameNotFoundException
- AccountBannedException
- UserIsAdminException

IAccountManager

Responsibility

- Act as an input boundary for controllers to pass user input data to the AccountManager use case

Collaborators

- IncorrectPasswordException
- UsernameExistsException
- UsernameNotFoundException
- AccountBannedException
- UserIsAdminException

Controllers

SignUpController

Responsibilities

- Store IAccountManager
- Store AccountRequestFacade
- Store ISleeper
- Read username and password input from user to process sign up
- Send username and password data to IAccountManager to create a new user (if possible)
- Catch related exception and output failure message for sign up
- Output success message for sign up
- Redirect user to the account request page upon successful sign up

Collaborators

- IAccountManager
- AccountRequestFacade
- ISleeper (implemented by Sleeper)
- UsernameExistsException

LoginController

Responsibilities

- Store IAccountManager
- Store AdminRequestFacade
- Store AccountRequestFacade
- Store ISleeper
- Read username and password input from user to process login
- Send username and password to IAccountManager to verify login credentials
- Catch related exceptions and output failure message for login
- Output success message of user login
- Upon successful login, redirect users to the admin request page or the account request page depending on whether the account is an admin or regular account

Collaborators

- AdminRequestFacade
- AccountRequestFacade
- IAccountManager
- ISleeper (implemented by Sleeper)
- IncorrectPasswordException
- UsernameNotFoundException
- AccountBannedException

AdminManagementController

Responsibilities

- Store IAccountManager
- Store ISleeper
- Read input from admin for the username of an account that will be promoted to admin
- Send username of the account to be promoted to IAccountManager which handles promotion logic
- Catch related exceptions and output failure for account promotion
- Output success message for account promotion
- Read input from admin for username and password of a new admin account to be created
- Send username and password of the admin account to be created to IAccountManager which handles admin creation logic
- Catch related exceptions and output failure for admin creation
- Output success message for admin creation

Collaborators

- ISleeper (implemented by Sleeper)
- IAccountManager
UsernameNotFoundException
- UserIsAdminException
- UsernameExistsException

BanUserController

Responsibilities

- Store IAccountManager
- Store ISleeper
- Read input for the username of an account to be banned
- Send the username of the account to be banned to IAccountManager which handles account banning logic
- Catch related exceptions and output failure message for banning an account
- Output success message of banning an account
- Read input for the username of an account to be unbanned
- Send the username of the account to be unbanned to IAccountManager which handles account unbanning logic
- Catch related exceptions and output failure message for unbanning an account
- Output success message for unbanning an account

Collaborators

- IAccountManager
- ISleeper (implemented by Sleeper)
- UserIsAdminException
- UsernameNotFoundException

DeleteUserController

Responsibilities

- Store IAccountManager
- Store ISleeper
- Read input for the username of an account to be deleted
- Send the username of the account to be deleted to IAccountManager which handles account deletion logic
- Catch related exceptions and output failure message for account deletion
- Output success message for account deletion
- Read input for users deleting their own accounts
- Send the account's username to IAccountManager which handles self-deletion logic
- Output success message of deleting the user's own account

Collaborators

- IAccountManager
- ISleeper (implemented by Sleeper)
- UserIsAdminException
- UsernameNotFoundException

LogoutController

Responsibilities

- Store ISleeper
- Output that the log out was successful

Collaborators

- ISleeper (implemented by Sleeper)

ViewHistoryController

Responsibilities

- Store IAccountManager
- Store ISleeper
- Output the log in history of the user who made a request to view his or her log in history

Collaborators

- IAccountManager
- ISleeper (implemented by Sleeper)

AccountRequestFacade

Responsibilities

- Store IAccountManager
- Store ISleeper
- Store controllers like DeleteUserController, LogoutController, ViewHistoryController
- Present a CLI allowing users to input a request they wish to perform
- Read user's request for commands they wish to perform
- Call the correct controller to handle the user's request

Collaborators

- IAccountManager
- ISleeper (implemented by Sleeper)
- DeleteUserController
- LogoutController
- ViewHistoryController

AdminRequestFacade

Extends AccountRequestFacade

Responsibilities

- Store BanUserController and AdminManagementController
- Extend functionalities of AccountRequestFacade
- Present a CLI allowing admins to input a request they wish to perform
- Read admin's request for commands they wish to perform
- Call the correct controller to handle the admin's request

Collaborators

- AccountRequestFacade
- IAccountManager
- ISleeper (implemented by Sleeper)
- DeleteUserController
- LogoutController
- ViewHistoryController
- BanUserController
- AdminManagementController

Gateways

JsonReader

Implements: IJsonReader

Responsibility

- Read the JSON object stored at the file found at the input file path

Collaborations

- IJsonReader

IJsonReader

Responsibility

- Offer a service to read JSON object stored at the file found at the input file path

Collaborations

JsonWriter

Implements: IJsonWriter

Responsibility

- Write an input JSON object to a file found at the input file path

Collaborations

- IJsonWriter

IJsonWriter

Responsibility

- Offer a service to write an input JSON object to a file found at the input file path

Collaborations

JsonToObject

Implements: IJsonToObject

Responsibilities

- Convert a JSON object to a Java object

Collaborators

- IJsonToObject

IJsonToObject

Responsibilities

- Offer the service to convert a JSON object to a Java object

Collaborators

ObjectToJson

Implements: IObjectToJson

Responsibilities

- Convert a Java object to a JSON object

Collaborators

- IObjectToJson

IObjectToJson

Responsibilities

- Offer a service for converting a Java object to a JSON object

Collaborators

PasswordHash

Implements: IHash

Responsibilities

- Return the SHA-512 hash of an input string

Collaborators

- IHash

IHash

Responsibilities

- Offer the service of returning a hashed string for an input string

Collaborators

JsonParser

Implements: IJsonParser

Responsibilities

- Parse JSON in the format of a string into a JSON object

Collaborators

- IJsonParser

IJsonParser

Responsibilities

- Offer the service of parsing JSON in the format of a string into a JSON object

Collaborators

Sleeper

Implements: ISleeper

Responsibilities

- Pause the program for a duration specified as an input in milliseconds

Collaborators

- ISleeper

ISleeper

Responsibilities

- Offer the service of pausing the program for a duration specified as an input in milliseconds

Collaborators

Exceptions

AccountBannedException

Responsibilities

- Throw an exception related to an account being banned

Collaborators

IncorrectPasswordException

Responsibilities

- Throw an exception related to a password being incorrect for an account

Collaborators

UserIsAdminException

Responsibilities

- Throw an exception related to a user being an admin

Collaborators

UsernameExistsException

Responsibilities

- Throw an exception related to a username being taken by another account

Collaborators

UsernameNotFoundException

Responsibilities

- Throw an exception for a username not matching the username of any existing accounts

Collaborators