Tech Design

Problem

As a developer, I need to create an application that can store members within a key, value system accessible via a command prompt interface.

Notes

- Technologies
    - Java
    - Lombok
    - No UI required.
    - No database required.
- Purpose
    - Application to add, remove, and lookup members within a key, value system.
    - Users can:
        - Add members to keys. (ADD <KEY> <MEMBER>)
        - Remove members from keys. (REMOVE <KEY> <MEMBER>)
        - Lookup a list of keys. (KEYS)
        - Lookup members of a particular key. (MEMBERS <KEY>)
        - Remove all members from a key. (REMOVEALL <KEY>)
        - Clear all members and keys. (CLEAR)
        - Check if a key exists. (KEYEXISTS <KEY>)
        - Check if a member exists within a key. (MEMBEREXISTS <KEY> <MEMBER>)
        - Get all members from all keys. (ALLMEMBERS)
        - Retrieve a list of all “items” (key/values within the system). (ITEMS)

Solution

- Preface
    - As the position in question and my prior experience leans towards backend experience, the approach used for this project is to structure the application like a backend API.
    - The DB component in this case will be substituted with a class called “ItemRepository.java” to simulate a DB that the API would interface with.
- Controller
    - A central controller called “ItemController.java” will handle the following:
        - Prompting the user for input.
        - Parsing user input and performing basic validation.

        - Calling necessary services.
        - Handling any exceptions.
- Help Service
    - As an artifact of the limitations of this project, a “HelpService.java” class will be created to handle providing the user with input assistance.
- Item Service
    - An “ItemService.java” class will be created to handle processed requests from the “ItemController.java” class.
    - More specific validation for the individual requests will be handled in the services.
- Item Repository
    - An “ItemRepository.java” class will stand in for the DB component that will handle data access and basic data validation.