**Username List Problem**

Please provide a working (build and compiles) small App/Solution in Java that solves the following problems:

**Functional Requirements:**

The App receives a username from the user and it checks in a storage (could be anything DB, dictionary, file, mock data, etc.. whatever you prefer) if the username is already taken or if is a valid new username.

If the username is valid it returns a success Boolean result if not it returns a False Boolean result with a List of at least 14 alternate suggested usernames based of the input username string. The alternate usernames List should be in alphabetical order and also each suggested username should be unique. (e.g: input username is john and taken, the suggested alternates could be : john1, john2,johnJohn,etc)

There is also a dictionary/list of restricted words which the username should not contain. This dictionary of words could contain any list of words that we want to exclude from being used in the username. So a username would fail also if it contains a restricted word and it would also generate the alternate username list from this. (e.g: possible List : cannabis, abuse, crack, damn, drunk, grass.) The design should allow adding more restricted words to the list in the future in an easy manner.

The usernames in the suggested alternate list should also be available and should not contain a restricted word. **Hint:** The app should try to generate 14 possible usernames from these rules at least three times. It if it is unable to generate 14 usernames it returns the list with the number it could generate (less than 14).

Example of a possible interface

Result<Boolean, List<String>> result = checkUsername(String username)

Result contains the Boolean which identifies the result of the operation. TRUE= success: username is valid, FALSE = fail: username is invalid.

**Input validation:**

The username should be at least 6 characters long or else throw an exception.

**Technical Requirements:**

Performance and maintainability should be considered as part of the design.

Unit tests of the main functionality at least to be able to test two usernames in a row. (e.g john, john, john1, john1)

**Nice to Have (optional):**

Send Github URL of the project.

Use Spring framework to wire dependencies if necessary.