Plan for Coding in Spark

Friend Recommendation

1. Extract <User> and <Connections> from the input file.
2. Separate <Connections> by comma and save <Connections> as a list.
3. Map input, <User> and <Connections>, to (userID, friendID). At this point, userID is the key and friendID is the value, which is a list of user IDs.
4. For a same userID, map every friend pair (friend 1 & friend 2) in FriendID as ((friend1, friend2), 1). Here, friend pair (friend1, friend2) is the key while "1" is the value.
5. For each key (friend1, friend2), sum the values "1" to get the number of occurrences of different friend pair. At this point, we have ((friend1, friend 2), count), where "count" represents the number of occurrences of each friend pair.
6. Flip the key-value pairs to (friend1, (count, friend2)). Now, friend1 is our key and (count, friend2) is the value.
7. For each key friend1, sort the value (count, friend2) by "count" in descending order, and just keep the first ten values. These are the largest 10 users that are not already connections with the key friend1. At this point, the structure of key-value pair is still (friend1, (count, friend2)).
8. Flip the structure to (friend1, friend2). At this point, we get rid of the "count" since it is not required in the output.
9. For each key friend1, put the values in a comma-separated list as the <Recommendations> of this key. Then, print and save a line per user in the format: friend1<tab>friend2_list. Here, friend2_list represents the comma-separated list of <Recommendations> we just created.
10. Now, we obtain the output required. friend1 is the <user> and friend2_list is the <Recommendations>.