Speaker ID Complete Tests

Description

There're three complete test cases to test the speaker ID on large inputs, as follows:

1. **Case1:** large number of samples, small sample size (~15 sec)
   - 11 users (each with ~100 training samples and 40 testing samples)
2. **Case2:** small number of samples, medium sample size (~2 min)
   - 11 users (each with ~10 training samples and 4 testing samples)
3. **Case3:** small number of samples, large sample size (~6 min)
   - 11 users (each with 2 training samples and 1 testing sample)

Each case is divided into training and testing sets. You need to

1. load the training set of all users *(using a function that is described below)*
2. Extract the features of all training samples for all users and save them
3. load the testing set of all users *(using a function that is described below)*
4. compare one (or more) of the testing samples with all training samples and find its matched user

> **IMPORTANT NOTE: before the final delivery**, for each case: you need to extract the features from training set and save them to file/DB. Your project should be able to load the saved features of a specific case from file/DB.

Steps

1. Download the speaker ID database from the following link:
   https://drive.google.com/file/d/0B3M4t0f9neFTWU0yN1U0Zmt4bjA/view?usp=sharing
2. Add the "TestcaseLoader.cs" to your project
3. To load the **training samples** of all users in your code, call the following function:

   List<User> LoadTestcase**N**Training(string trainingListFileName)

   Where:
   - **N**: is the test case number (1, 2 or 3)
   - trainingListFileName: complete path of the "TrainingList.txt" from the downloaded database
   - List<User>: type of the returned parameter, containing a list of User objects, each object contains:
     1. string UserName: user name
     2. List<AudioSignal> UserTemplates: list of the training signals for this user. The silence is already removed from each signal.

4. **Extract the features** from the loaded training samples and save them.
5. To load the **testing samples** of all users in your code, call the following function:

   List<User> LoadTestcase**N**Testing(string testingListFileName)

   Where:
   - **N**: is the test case number (1, 2 or 3)
   - testingListFileName: complete path of the "TestingList.txt" from the downloaded database
   - List<User>: type of the returned parameter, containing a list of User objects, each object contains:
     1. string UserName: user name
     2. List<AudioSignal> UserTemplates: list of the testing signals for this user. The silence is already removed from each signal.

6. **Compare ONE (or more) of the testing samples** with all training samples, then, add the name of the matched user to a list of strings (List<string>).