**Short summary of project (re-cap one paragraph)**

This tool is intended to analyze a Twitter user’s general mood.  It is intended to be used *in addition to* a human analysis component.  Its purpose is to provide a basic overview of a user’s average mood, with the knowledge that human behavior is difficult to perfectly categorize, and therefore the results should be taken with a grain of salt. The project will take a sample tweet or a real tweet and then perform sentiment analysis on it. After many tweets have been analyzed, a summary of the Twitter user will be generated.

**Status report (multiple paragraphs)**

Austin is working on using the Twitter API and connecting to it using a Twitter4J library. This is providing us with access to user’s past tweets. This requires a developer Twitter account which can easily be created. Then access keys have to be generated in order to grab any information from Twitter. Some functionality of the Twitter4J library was tested such as the “getUserTimeline()” method. We managed to return 100 most recent tweets from the CNN news account.

Christian is working on the database. This portion is going to house all of our twitter data. This allows us to break things up into the specific parts we need. It also is very handy in holding all the information we want to excess in one place. The database model is highly similar to the object model (and having an imperfect object model means we must be very careful about how everything gets integrated and a few changes are being made as we go). Christian got the basics of the database setup, and also got a good JDBC connection from the program. From here, we just need to write several helper functions that can be called (most likely statically) as needed.

Tony has been working on the test data we are going to use to test out our program. This data is then going to be transferred into our database so we can check how our program works and handles data. Once we get the results of this and figures out how everything works we then can use real twitter data to work on the more complex portions of the project. We have also gotten people to donate tweets to our project. This will be helpful to have since we know who they are coming from.

Thien is working on getting the general framework to the project and setting up classes and organizing how they will relate to each other.

Andrew has been setting up the stanford NLP libraries, which will contribute to our analysis of a person’s mood. This library helps to determine if it is positive or negative but our plan is to later integrate a more diverse moods than these. He has been also researching how we can use the NLP tools to accomplish this. Andrew also collected permission from various twitter users to use their real tweets in our project.

What we have left to do is bring all these things together:

* We have JDBC connectivity in Java but are not using the data from the DB, this will have to be added.
* We also have the ability to pull from Twitter but we have not created a function to put it into a database to be used in our program.
* We have the NLP libraries imported and a general Sentiment analyzer working for plain text, but have yet to expand this and make it more specific sentiments.

**Problems, risks and strategies**

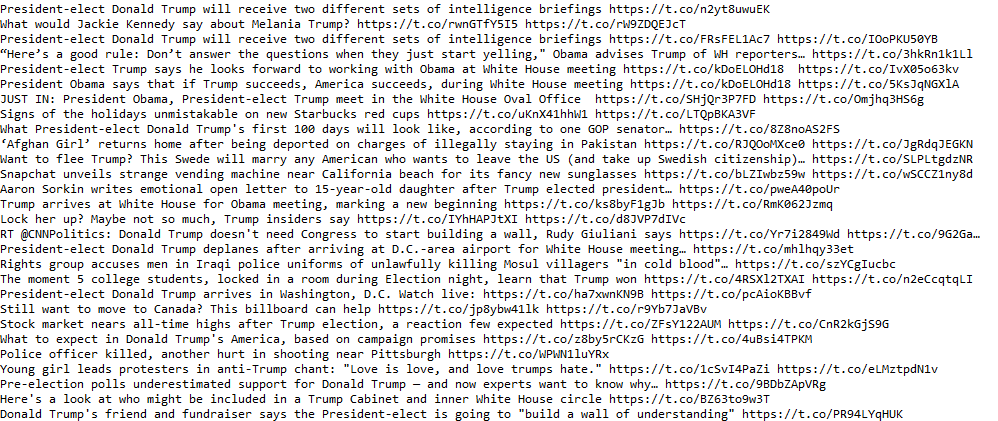
A problem that this project might encounter is being unable to access Twitter timelines. If the Twitter4J library for some reason stops working, then we have to rely on the database’s sample tweets. Another issue we could have is with the database. If we end up setting one of the classes up wrong this could cause us problems. Our strategies for these is to have good error reporting and work more add making the class structure reflect our database.

We also want to make sure the we are using the right sentiment on people's tweets. This is something that is a little more complex so to start we have test data that we will evaluate our program on. Later we will use some real Twitter feeds (with permission) and evaluate those. If the Twitter users are willing to give feedback that would be a desired as well. One way we are helping to mitigate this risk is by looking at third-party tools and crafting test cases with known oracles to ensure we get accurate results.

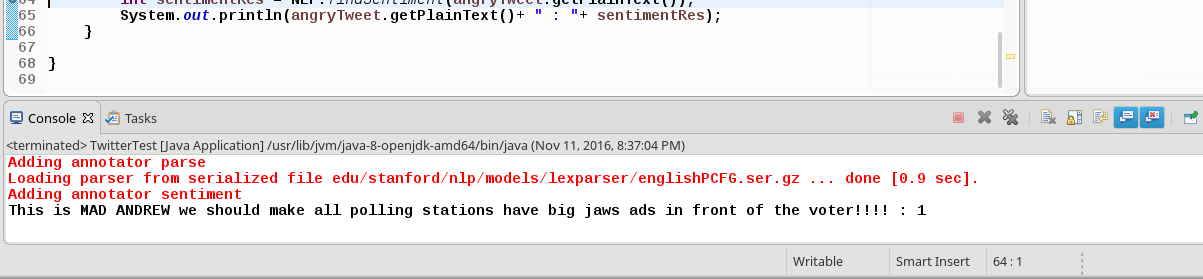
One other problem that arises to sentiment analysis is the way that sentences are structured and give itself meaning based off the grammar used. We are delving into Stanford's Natural Language Processing Libraries in order to hopefully solve this problem. They have a parser that will group certain words based on the context of the sentence. We are still researching this but it should prove effective.

Finally, one thing we must be very aware of is the integration step. We have taken to individually developing separate modules, but we will need to allocate a significant amount of time for getting together and figuring out how all of the separate pieces will work together. Additionally, we are working across several different development environments, which mean we each have to do a few things differently and be careful about what gets committed to git and how. Up until now we have been mostly working separately, so we need to set aside time specifically for integrating, perhaps where the code is up on one computer and all members of the group explain to the rest how their module works and how it can be integrated into the system.

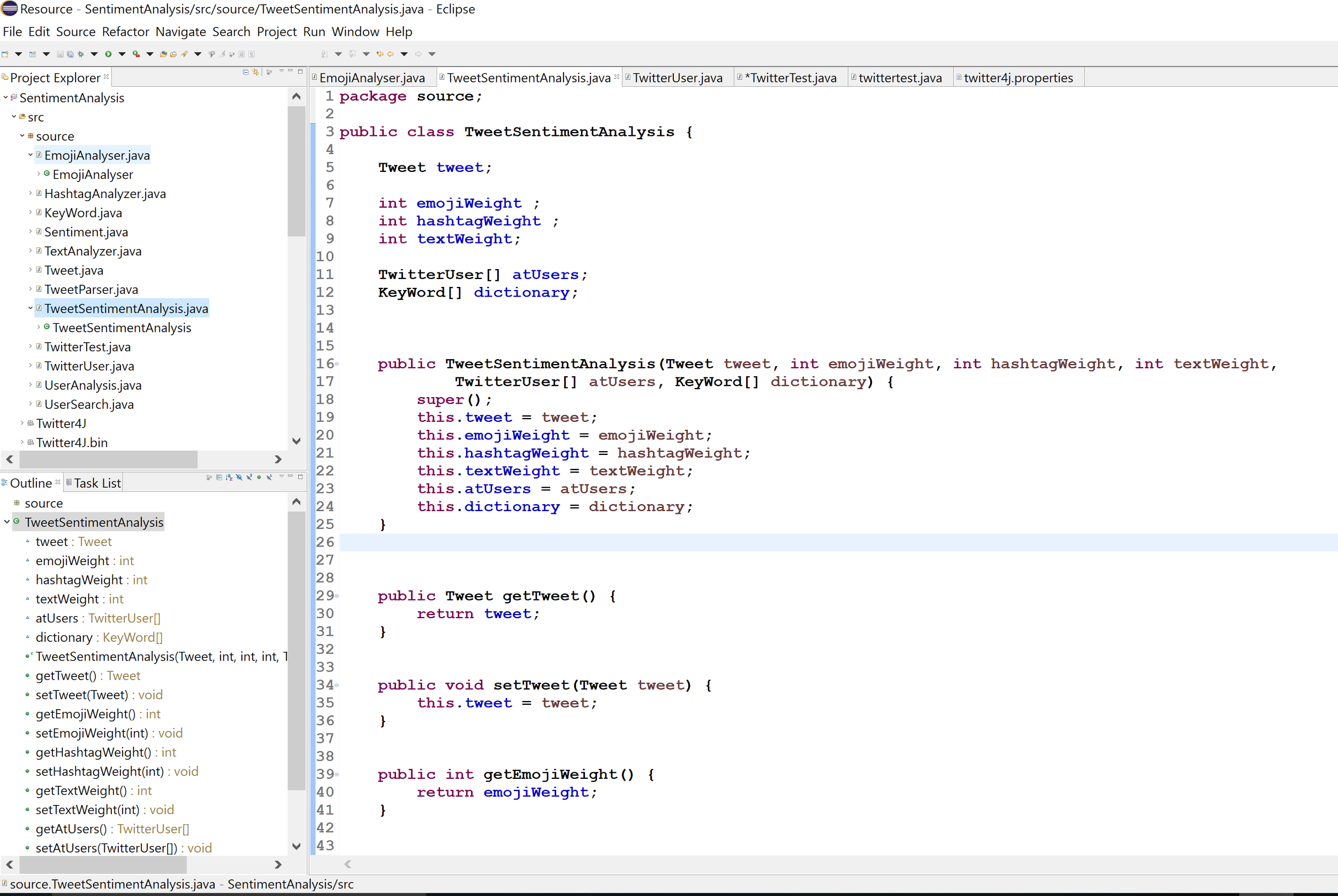
**Screen shots of prototypes and complete modules**

List of tweets grabbed from CNN’s timeline:

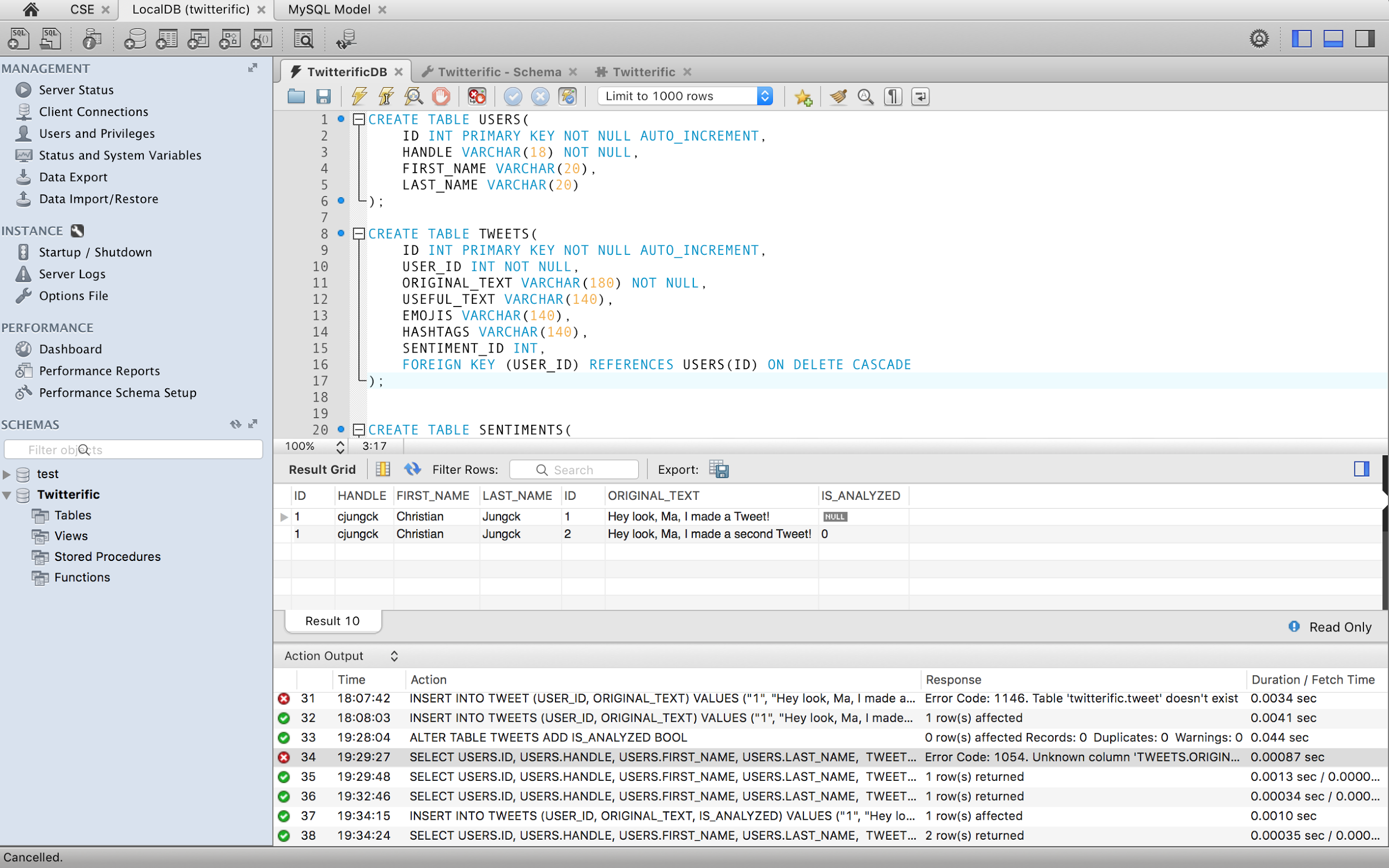
**Screenshot of an analysed tweet (plain text, lower score is more negative):**



**Screenshot of Java Classes:**



**Screenshot of Database structure:**



**Programmatic Database Connectivity:**

