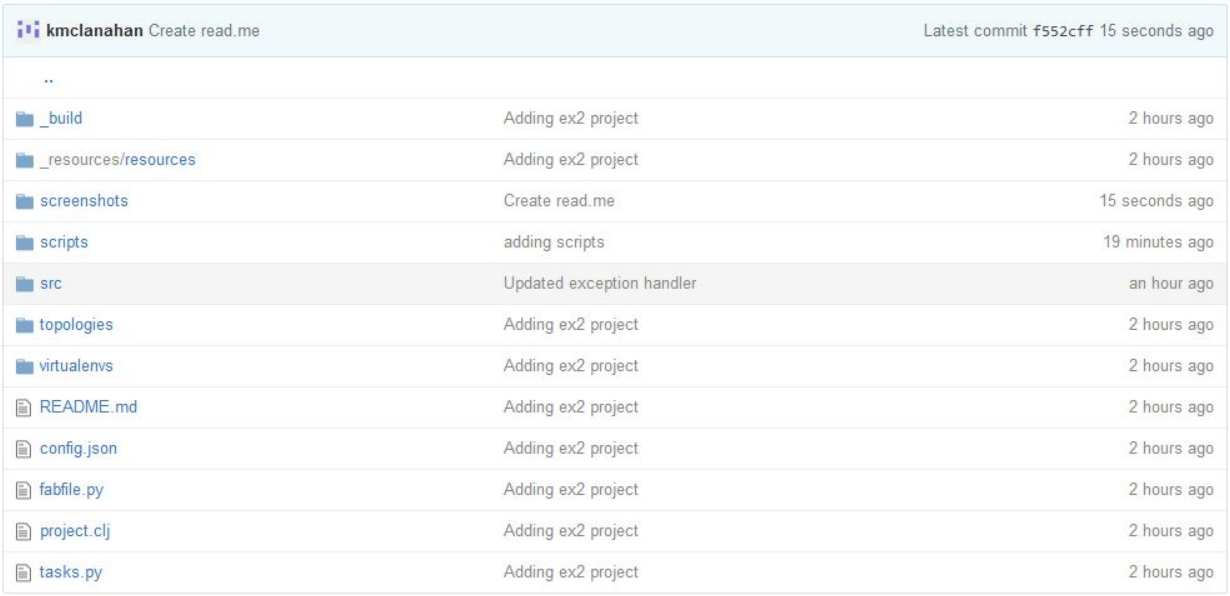
The followring is a screenshot showing the directory structure from my github repo:



kmclanahan Create read.me		Latest commit f552cff 15 seconds ago
..		
_build	Adding ex2 project	2 hours ago
_resources/resources	Adding ex2 project	2 hours ago
screenshots	Create read.me	15 seconds ago
scripts	adding scripts	19 minutes ago
src	Updated exception handler	an hour ago
topologies	Adding ex2 project	2 hours ago
virtualenvs	Adding ex2 project	2 hours ago
README.md	Adding ex2 project	2 hours ago
config.json	Adding ex2 project	2 hours ago
fabfile.py	Adding ex2 project	2 hours ago
project.clj	Adding ex2 project	2 hours ago
tasks.py	Adding ex2 project	2 hours ago

Most of this was setup via the streamparse framework by running ‘sparse quickstart …’ but the important directories are as follows:

/scripts: Contains finalresults.py and histogram.py
/src: Contains the spouts and bolts for the tweet word count application
/topologies: Contains the clojure topology for the tweet word count application
/screenshots: Contains the requested screenshots including a bar chart showing the top 20 words in my stream

Possible applications for this project:

1) Monitor the types of words associated with tweets around a specific brand to measure customer engagement, twitter ad campaigns, etc. in real-time
2) Monitor the stream for key disaster words like “earthquake” or “shooter” to detect emergencies in real-time.
3) Measure the relative popularity of celebrities over a period of time based on their appearance in tweets

Dependencies:
The UCB AMI running streamparse, postgres, and python 2.7.3
Clone of my github repo: https://github.com/kmclanahan/W205.git

Instructions to run the application are as follows:

1) Log into the UCB AMI as root
2) Mount a data volume that contains a postgres database called ‘tcount’ and a table named ‘Tweetwordcount’
3) Start that database using the start_postgres.sh script provided
4) Switch to the w205 user
5) Navigate to the .../W205/exercise_2/extwotweetwordcount directory in my github repo
6) Type ‘sparse run’ to run the application