Team Cacti - Emma, Yvonne, Angela, Wilson

Design Document: quenchr

Project Idea

Our project quenchr is a website that will help users quench their thirst for Likes and Followers by optimizing how and when they post. Quenchr will look at sentiment, content, length, and contained media (e.g links, images) of past posts on the Twitter account of a user to find what results in the most activity. It will also give the user the option to have our site post a given Tweet to their Twitter at the optimal time. We will accomplish this using data from the Twitter Search API (to get raw data about existing posts) and sentiment analysis with the Text Processing API.

JavaScript will be used to display explanatory graphs for the user.

Templates and Site Map

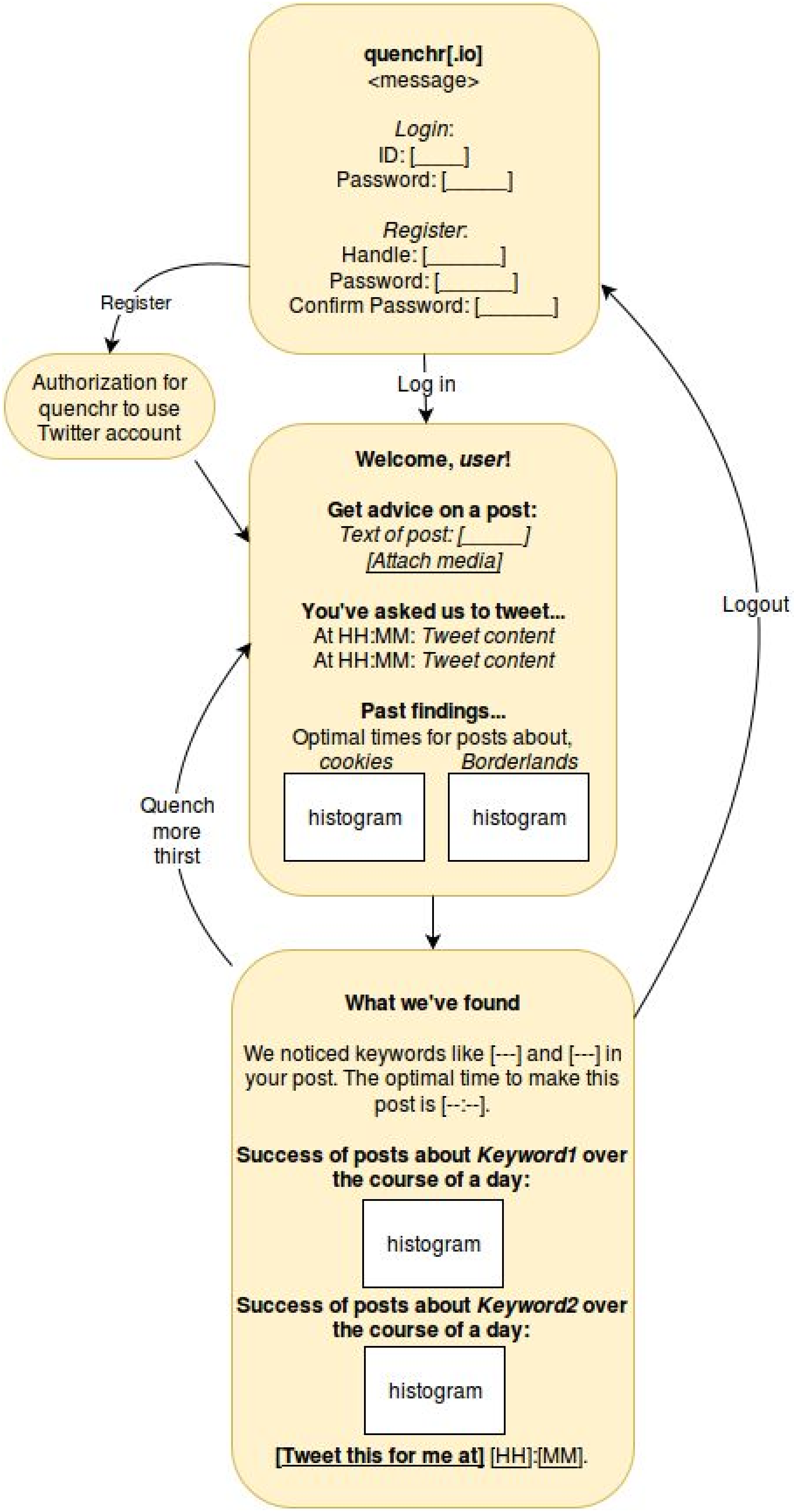
Templates

- layout.html -- banner with navigation links, with a `block` for specific content
- welcome.html -- for login and register
- dashboard.html -- user dashboard with information and forms to process a new Tweet

  Parameters:
  - list of dictionaries holding a “time”, “text” (of Tweet), and “media”, for displaying which Tweets will be automatically posted
  - list of dictionaries describing the graphs, with “keywords” and “hours” describing which keywords the graph reflects and which hours are best to post at
- results.html -- display information on the results of processing a potential Tweet

  Parameters:
  - list of important keywords (to display in the message at the top)
  - time at which it is best to post
  - list of dictionaries relating “keyword”s to how well those keywords do at various times in the day (for histograms)

Site map…



quenchr[.io]
<message>

Login:
ID: [____]
Password: [_____]

Register:
Handle: [______]
Password: [______]
Confirm Password: [______]

Authorization for quenchr to use Twitter account

Welcome, *user*!

Get advice on a post:
Text of post: [_____]
[Attach media]

You've asked us to tweet...
At HH:MM: *Tweet content*
At HH:MM: *Tweet content*

Past findings...
Optimal times for posts about,
cookies *Borderlands*
histogram histogram

What we've found

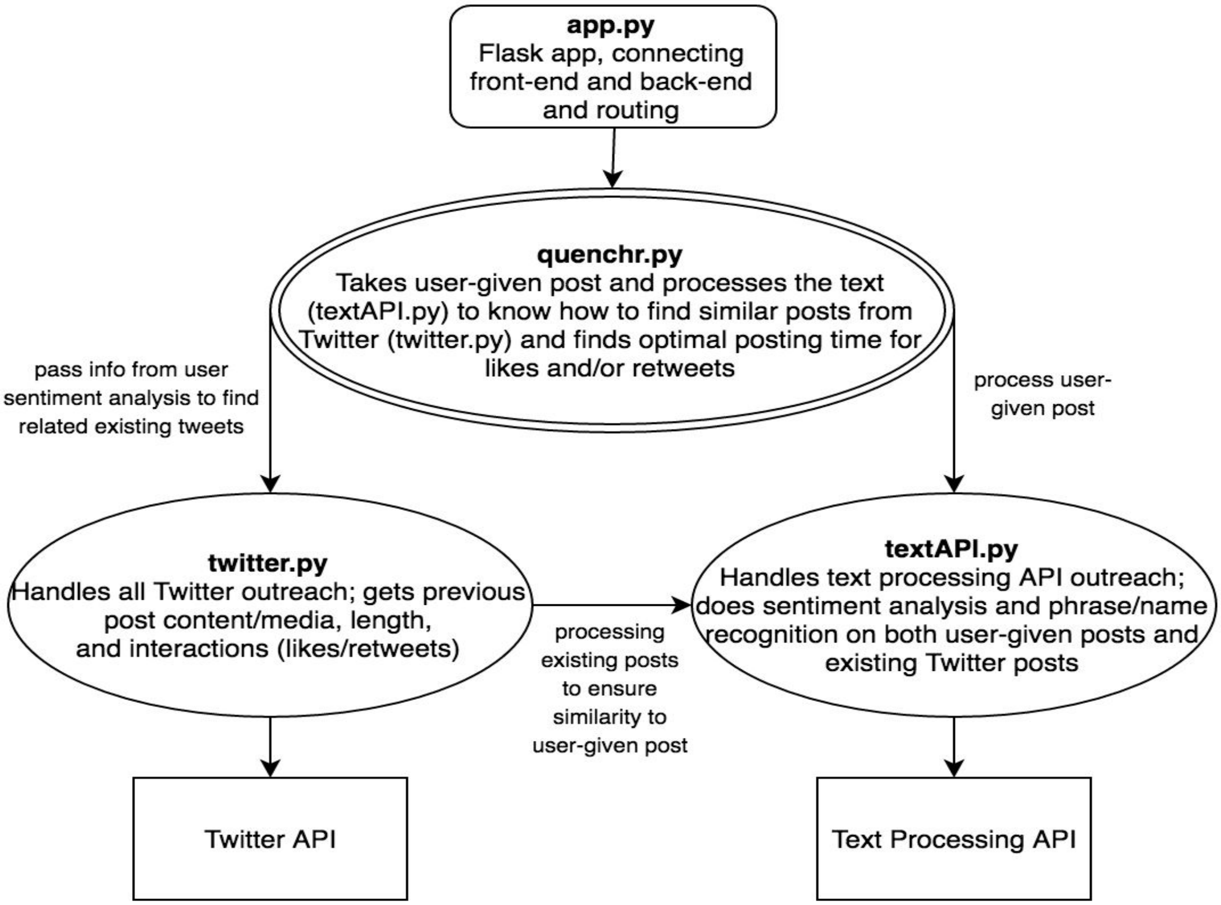
We noticed keywords like [---] and [---] in your post. The optimal time to make this post is [--:--].

Success of posts about *Keyword1* over the course of a day:
histogram

Success of posts about *Keyword2* over the course of a day:
histogram

[Tweet this for me at] [HH]:[MM].

Components


app.py
Flask app, connecting front-end and back-end and routing
quenchr.py
Takes user-given post and processes the text (textAPI.py) to know how to find similar posts from Twitter (twitter.py) and finds optimal posting time for likes and/or retweets
pass info from user sentiment analysis to find related existing tweets
process user-given post
twitter.py
Handles all Twitter outreach; gets previous post content/media, length, and interactions (likes/retweets)
processing existing posts to ensure similarity to user-given post
textAPI.py
Handles text processing API outreach; does sentiment analysis and phrase/name recognition on both user-given posts and existing Twitter posts
Twitter API
Text Processing API

Listing

- ❏ Python
  - ❏ Interfacing with APIs
    - ❏ **textAPI.py**: communicate with the Text Processing API
    - ❏ **twitter.py**: communicate with the Twitter API
  - ❏ **quench.py**: calculate the optimal post time, successfully quenching the user's thirst for likes
  - ❏ **app.py**: handle routing and templates
- ❏ Front End
  - ❏ / [root route]
    - ❏ **Before log-in:** description of website, and fields in which to enter the account name and the post to submit

- ❑ **Post login-in:** includes chrome for processing a new Tweet, a listing of Tweets the user has asked quenchr to schedule, and graphs from prior times the user processed potential Tweets.
- ❑ **After processing a potential Tweet:** a recommendation of when to post that Tweet, histograms which demonstrate why, and a form to have quenchr post the Tweet automatically at a given time (by default, the recommended time).

Database Schema

TABLE tweets

INT tweetID	TEXT handle	TEXT tweet	INT time

TABLE keywords

INT tweetID	TEXT keywords	INT time	INT strength

TABLE tweeters

TEXT handle	TEXT pass_hash	TEXT pass_salt	TEXT clientToken

Division of Labor

Project Manager: Angela
Backend / API handling: Yvonne
Backend / Processing API data: Emma (+ Everyone else for developing algorithm)
Frontend / Templates + CSS: Wilson
Frontend / JavaScript: Wilson

Timeline

1/6: Begin work and finalize design doc

1/9: Get login/register working, get API keys and preliminary info

1/11: Have Twitter API and Text Processing API setup and basic methods written

1/13: Have templates and layout of website setup and ready to display data from database

1/17: Complete algorithm for finding optimal time

1/19: Complete all database tables created and methods written, ready for testing results of algorithm

1/20: Complete the display of histograms for past and present posts

1/23: Use Twitter API to post for the user at the optimal time

For future if we have time:

- Retweeting other people’s posts to bump their popularity
- Most popular tweet of the day