

FUCK OFF AS A SERVICE

Goal: to allow users to tell people to “fuck off” in various ways

Implemented APIS:

- FOAAS
 - the pioneer API of our project. gives a fuck off message on request. nothing much other than this
- Twitter
 - required for OAUTH – is the way users will sign in and create accounts
 - allow users to tweet users fuck off messages
- Twilio
 - allow users to send fuck off text messages to other people
 - free trial: can only send users who “verify” their phone numbers, therefore not allowing users to send to anyone. NEED TO DECIDE IF WE WANT TO UPGRADE \$\$\$\$

Framework:

- Django
 - in python and running on our AWS EC2 instance
 - SEE TUTORIAL BELOW
- mySQL
 - required to maintain databases of users, their phone numbers, their preferences, etc

List of things we need to do:

Back End

Initialization

- set up twitter authentication
- set up chronic twilio jobs
- set up MySQL

Front End

- Login Page
- User’s profile page
 - Add a target phone number & select message customization
 - which messages
 - how often to send them
 - Add a twitter user
 - which messages
 - how often to send them & select message customization

EXAMPLE WEB PAGES

these are just to get an idea! and the ideas may change during development

welcome page

Welcome to FOAAS

Login with your twitter account

user

pass

after signing in

Logout

Your Profile:

Your registered numbers

999-999-9999

add a number

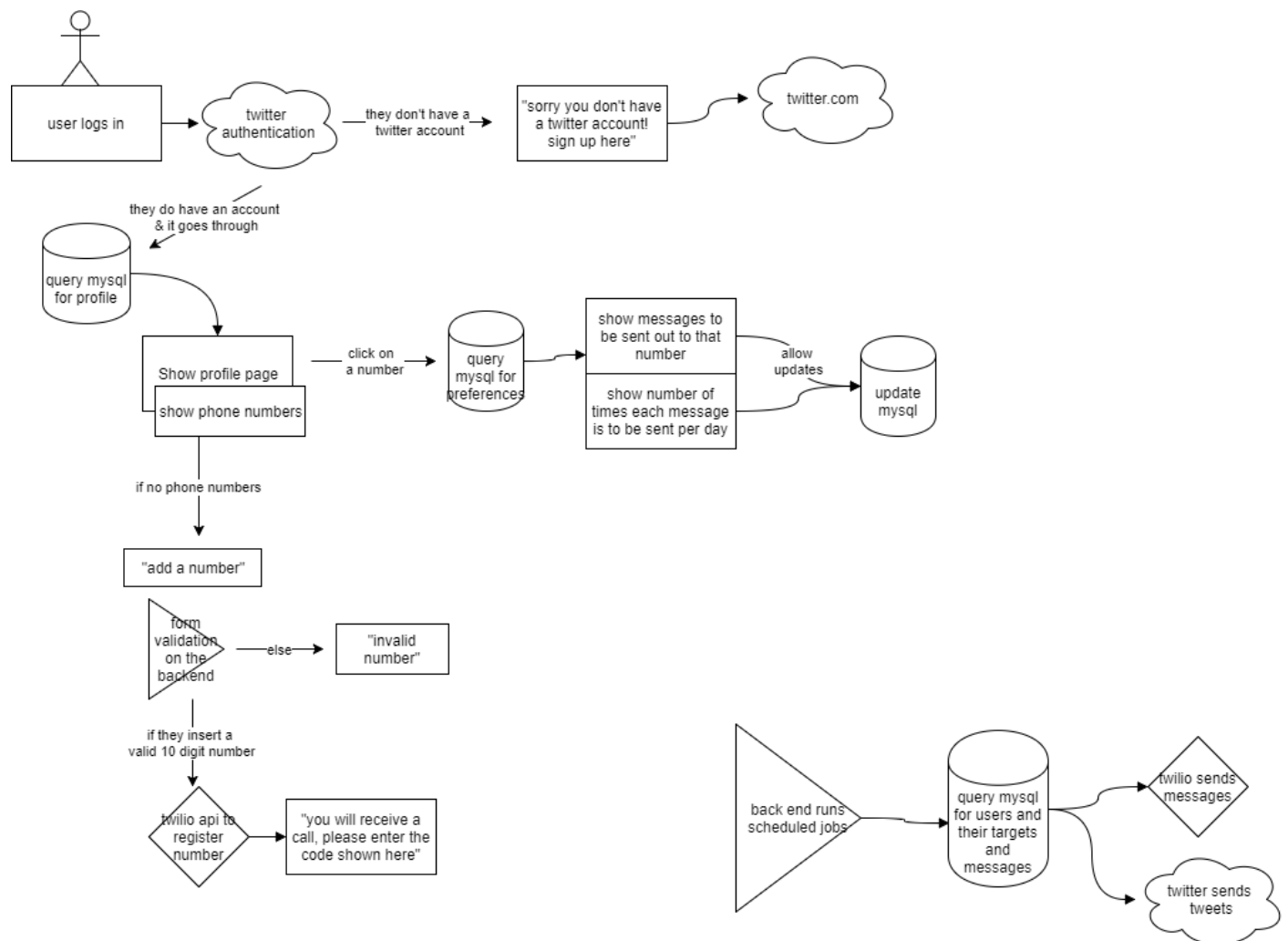
after clicking on a number

Logout

Your customization for 999-999-9999

messages	times per day
fuck	1
fuck	4
fuck	2

Example event flow with DJANGO & APIS working with FRONT END



MINI DJANGO TUTORIAL

- everything is in python
 - except for the templates which are html/css/javascript
- Django directory (just including the important stuff)

FOAAS

FOASS

settings.py

urls.py

FUCKOFF

apps.py

hello.html (front page) (this should be in templates, but it's here for now)

models.py

tests.py

urls.py (this is where we set up the urls)

views.py (this is where most of the work happens)

TEMPLATES

<html/css pages go here>

- FOAAS/FUCKOFF
 - this is the directory that we do most of the work for our app
 - view.py
 - where you write functions to handle things
 - handle POST/GET methods for API/html forms/etc
 - handles sql queries for updating the databases
 - handles which html pages are shown
 - urls.py
 - initialize the urls here
 - ex: /home, /user_profile
 - add the following to the list
 - url(<regex of the url destination (examples above)>, <function from views to show when you go to that url (remember that the views say which html is going to be shown)> <a name (which is optional)>)
- **GENERALLY you don't need to modify anything else**
 - changes to FOASS/settings.py or FOASS/urls.py will probably be necessary as we develop, but that will be because of configuration issues that we need to fix in order to get the app running smoothly.
 - you can *probably* ignore fuckoff/models.py, fuckoff/tests.py, and fuckoff/apps.py
 - this stuff is not necessary for the things we are doing
- the most important thing is that **EVERYTHING IS TIED TOGETHER**
 - you need html pages for you views to do/display anything

- you need view functions for you urls to do/display anything
 - everything is managed and controlled (invisibly) by FOASS/settings.py and FOASS/urls.py
- RUNNING THE SERVER
 - when you want to be able to go to ec2-52-43-159-58.us-west-2.compute.amazonaws.com in your browser you must do **python manage.py runserver 0.0.0.0:80**
 - then you are able to go to it in your browser and interact with the app
- PLEASE VIEW THE DOCUMENTATION AT
 - <https://www.djangoproject.com/start/>
 - <https://docs.djangoproject.com/en/1.11/>
 - Django has EXCELLENT documentation and because it's a very popular framework GOOGLE your questions and more than likely someone has already answered it on STACKOVERFLOW or similar forums
 - or you can ask me! chances are I may have encountered the problem before or I can help you search for a solution.