Requirements:

- MongoDB
- Browser (e.g. Chrome, Firefox), Javascript enabled
- Python 2.7.6
- Python-Packages:
    - BaseHTTPServer
    - os
    - base64
    - json
    - bson
    - pymongo
    - datetime
    - SocketServer
    - threading
    - collections
    - numpy
    - re
    - sys

Quick-Start:

1) Ensure all requirements are satisfied;
2) Populate your MongoDB with a database and the provided sentiment example collection;
3) Specify the *mongodb_settings* in the configuration.json;
4) Run “python annotationserver_multi_threaded_consolidation.py” in your terminal;
5) Open your browser and connect to localhost:8080;
6) Log-in using username “User1” and password “passwordxx1”.

Personalized Set-up:

1. Populate your MongoDB with the desired database and collection. For details on the requirements of each mode, please see section **database specifications** in the configuration_manual.pdf;
2. In the configuration.json, section the *accounts* (line 3), ensure that *usernames* and *passwords* are properly set;
3. Define the *annotation_mode* in configuration.json [bi-classification, tri-classification, textual-annotation, sentiment, sentiment-news], default: sentiment;

4. Depending on 3., modify the corresponding section (bi-classification in line 54 for mode *bi-classification;* further explanations are in the *configuration_manual.pdf*)

5. Create your own HTML FAQ and annotation/consolidation guidelines; place these in the folder *resources/individual*.

    5.1. annotation FAQs: *faq.html*;

    5.2. consolidation FAQ: *cfaq.html*;

    5.3. annotation guideline: *guide.html*;

    5.4. consolidation guideline: *guide.html*

    5.5. If images are used, place these in the folder *resources/individual/images*

6. Run “python awocato_server.py” in your terminal;

7. Open your browser and connect to localhost:8080 (if executed on a server, server_IP:8080);

8. Log-in using the defined username(s) and password(s).