Introduction to MongoDB

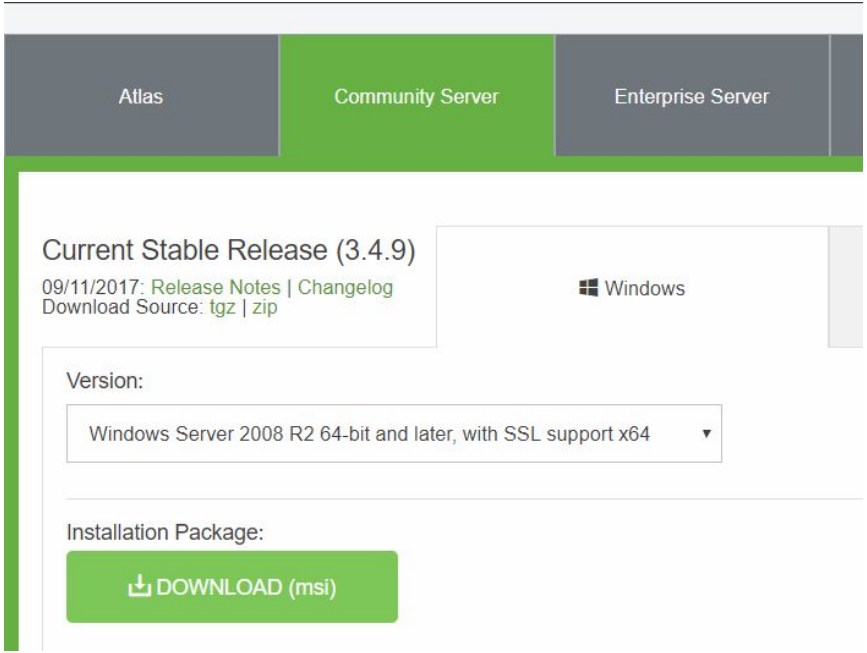
26/03/2018

What is MongoDB

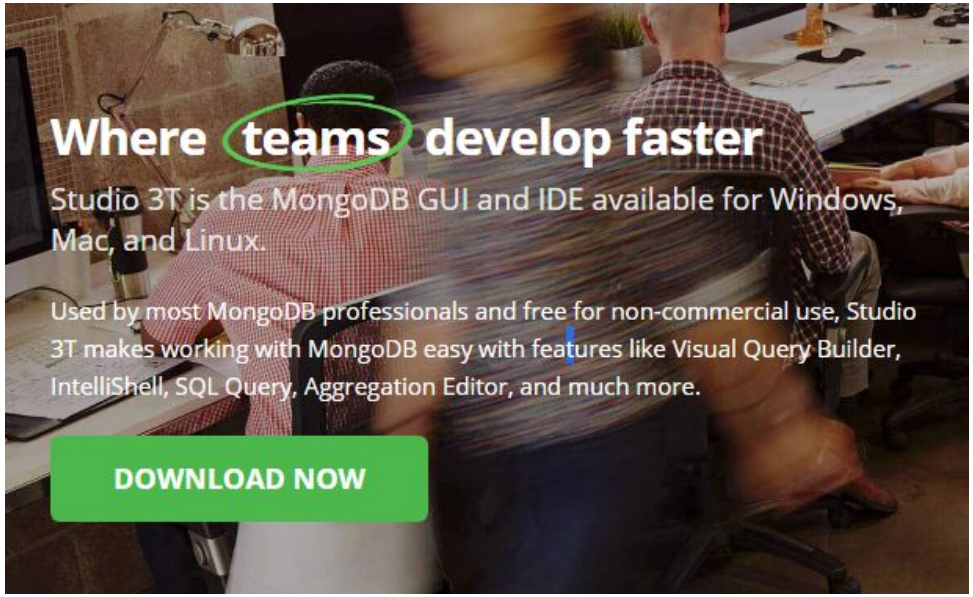
- It’s a free NoSQL document oriented database.
  - Not only SQL, since SQL queries are supported.
- It uses JSON-like documents to save data.
  - Flexible schemas - Allows missing fields.
- High performance, high availability, and automatic scaling.
  - Used by large companies like Amazon.

Installation

MongoDB community server

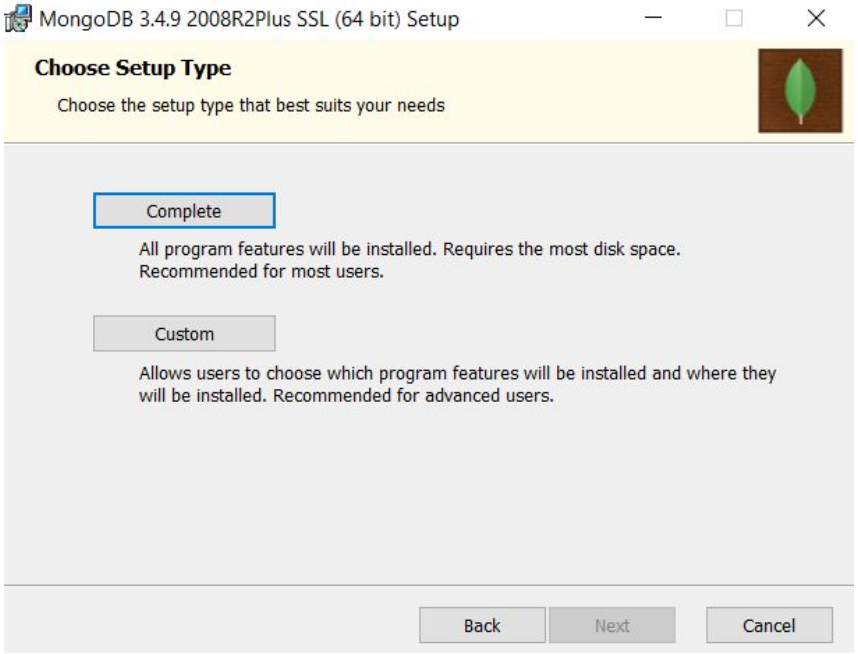


Studio 3T

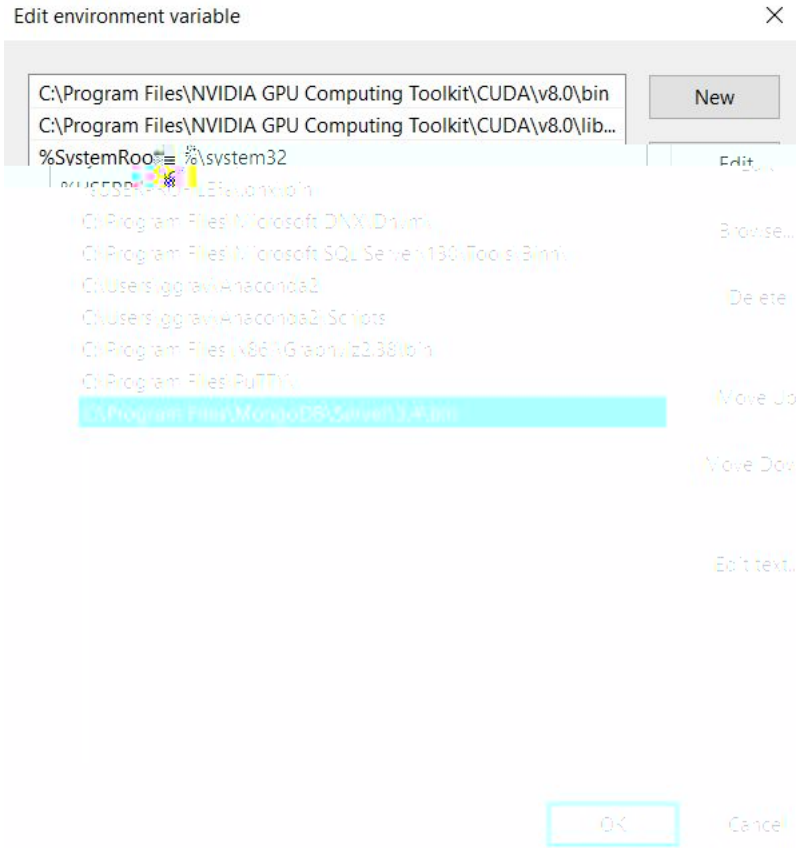


Installation

Step 1



Step 2



What is a document

- A sum of key / value pairs

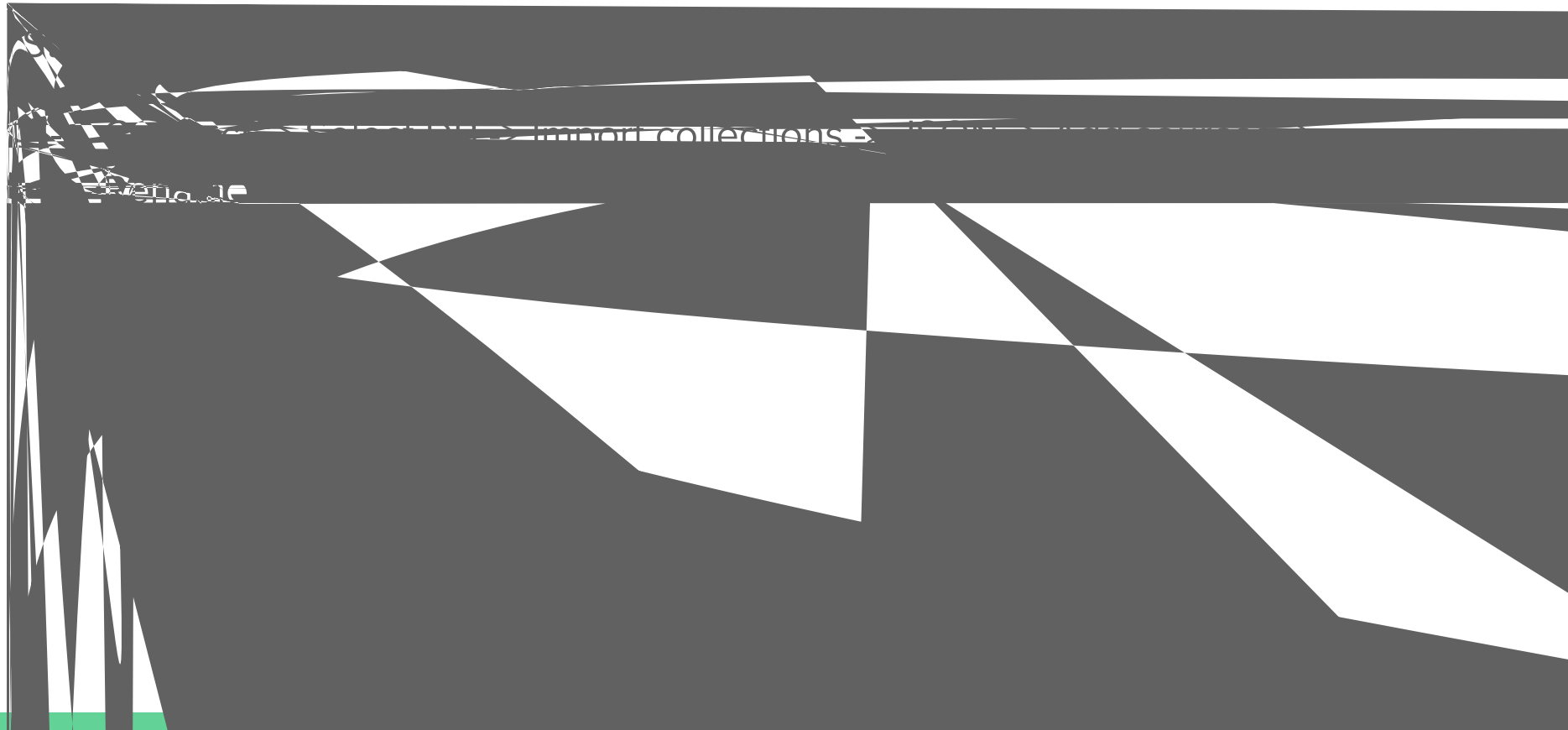
What is a collection

Operations

- Run / Connect to mongo
- Import documents
- Insert documents
- C

Connection

Import dataset



Insert documents

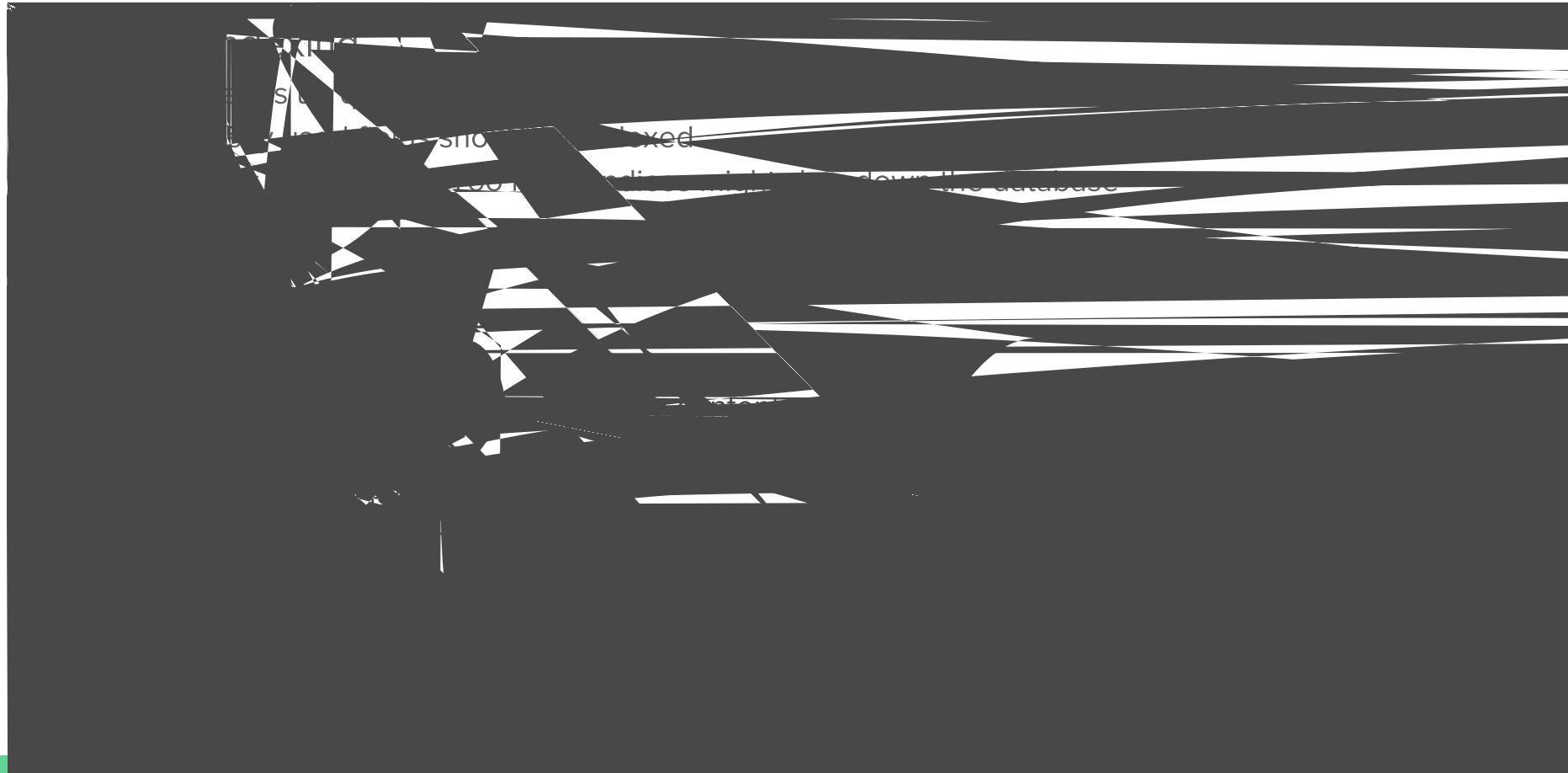
Python:

```
38          # Find the coordinates for each restaurant and
39          # save them to an external collection
40          all_restaurants = find_all_restaurants()
41          for restaurant in all_restaurants:
42              json_obj = {
43                  'name': restaurant['name'],
44                  'business_id': restaurant['business_id'],
45                  'longitude': restaurant['longitude'],
46                  'latitude': restaurant['latitude']
47              }
48              insert_to_db(json_obj, 'restaurants_coordinates')
```

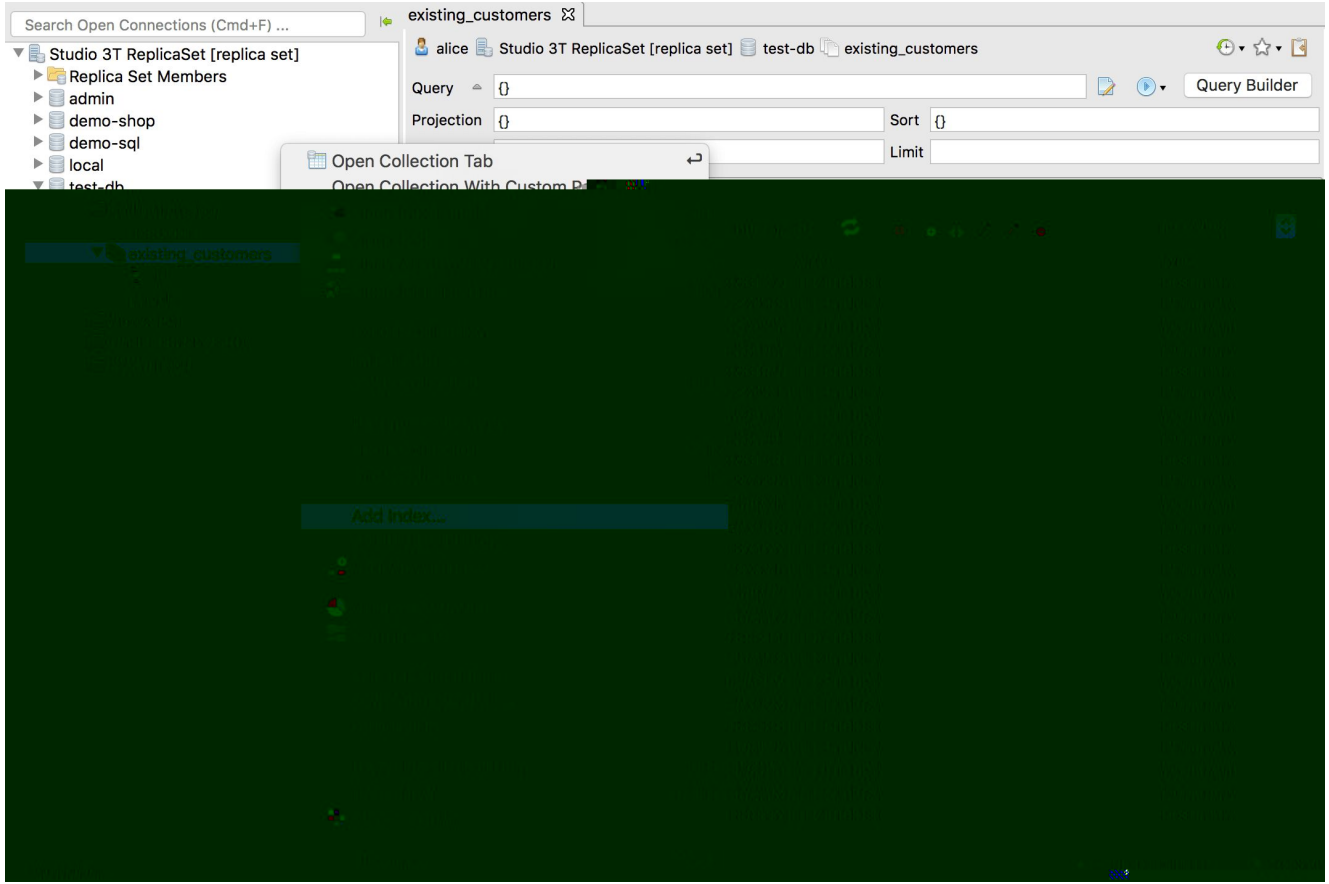
Data lookups - For quick data checking

"this lecture isn't mandatory, however it will offer you valuable informa-"

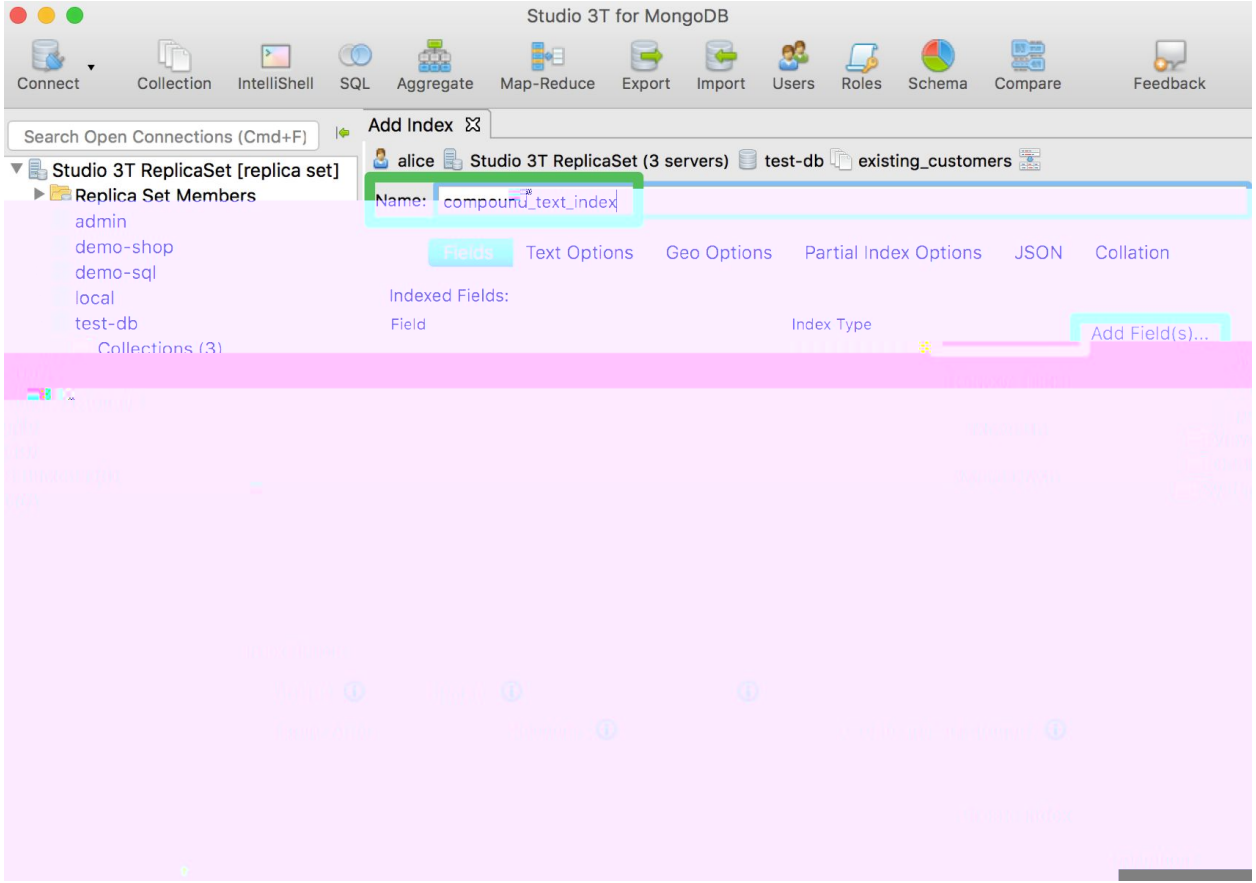
Indexing



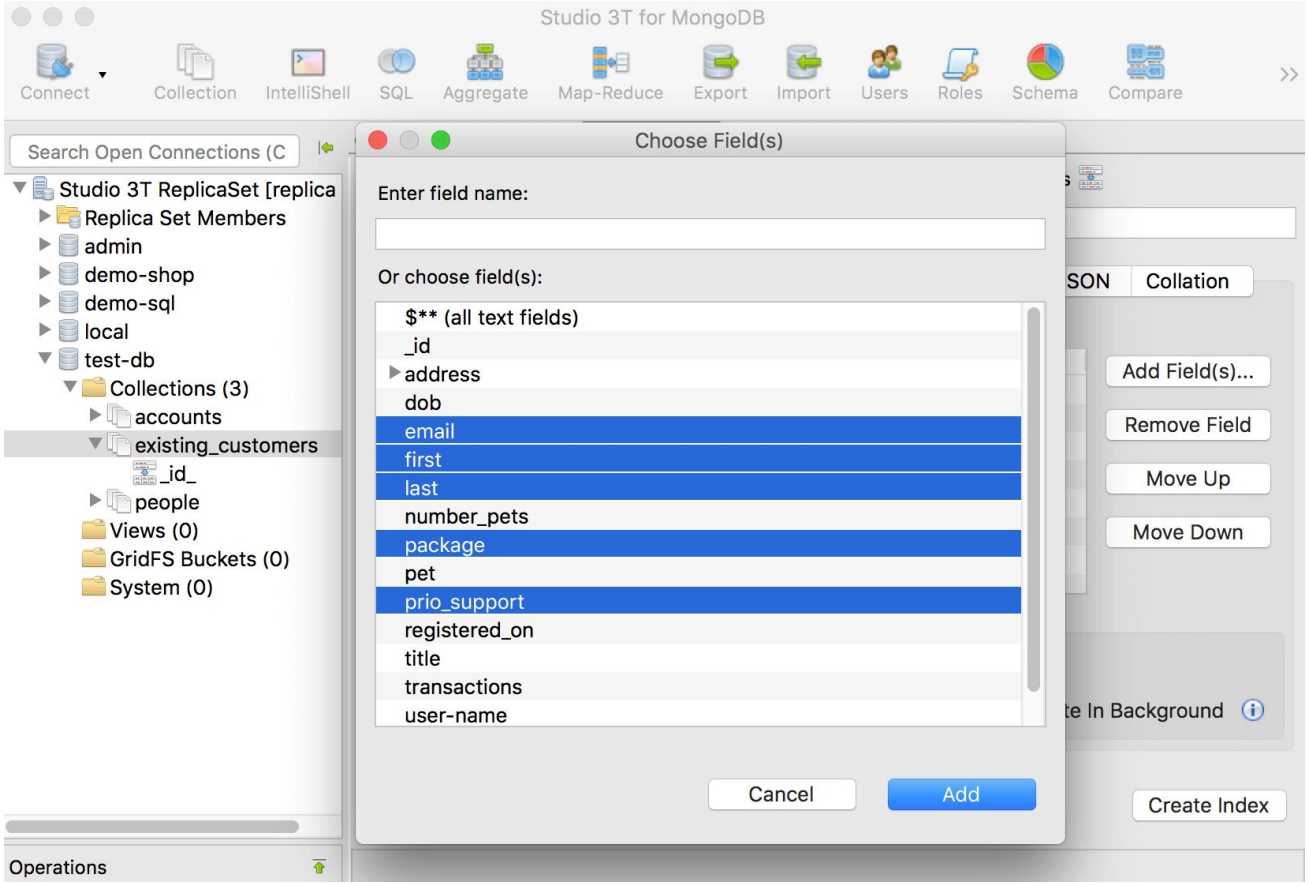
How to - Text indexing



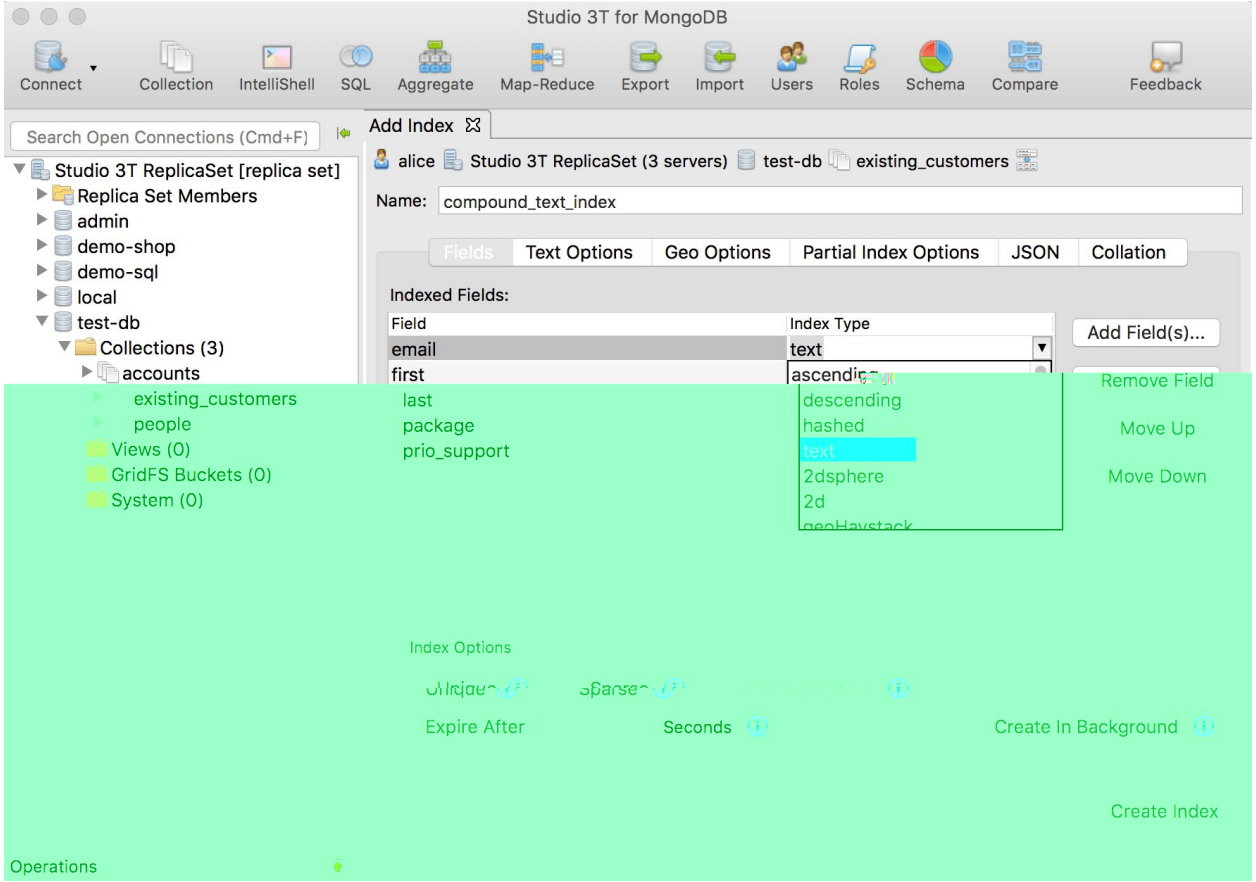
How to - Text indexing



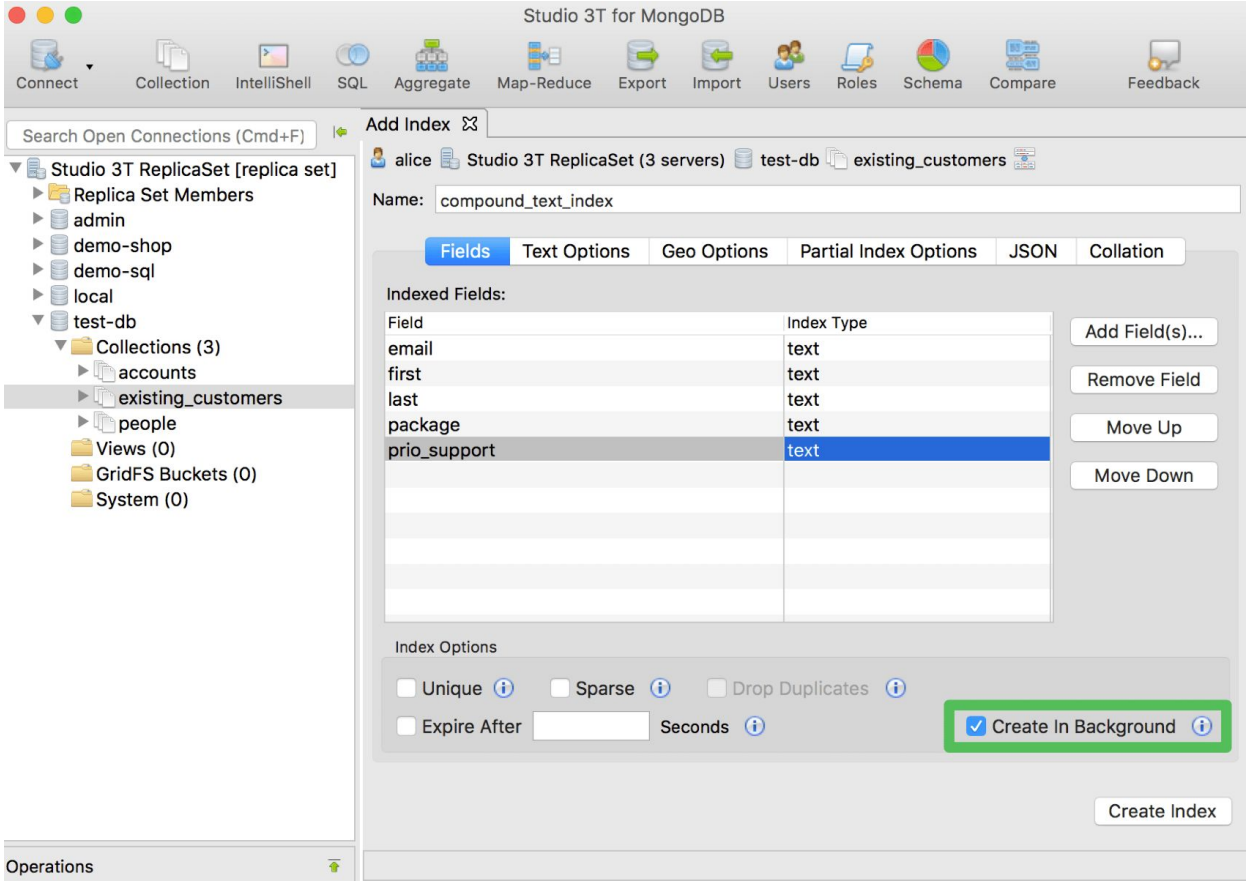
How to - Text indexing



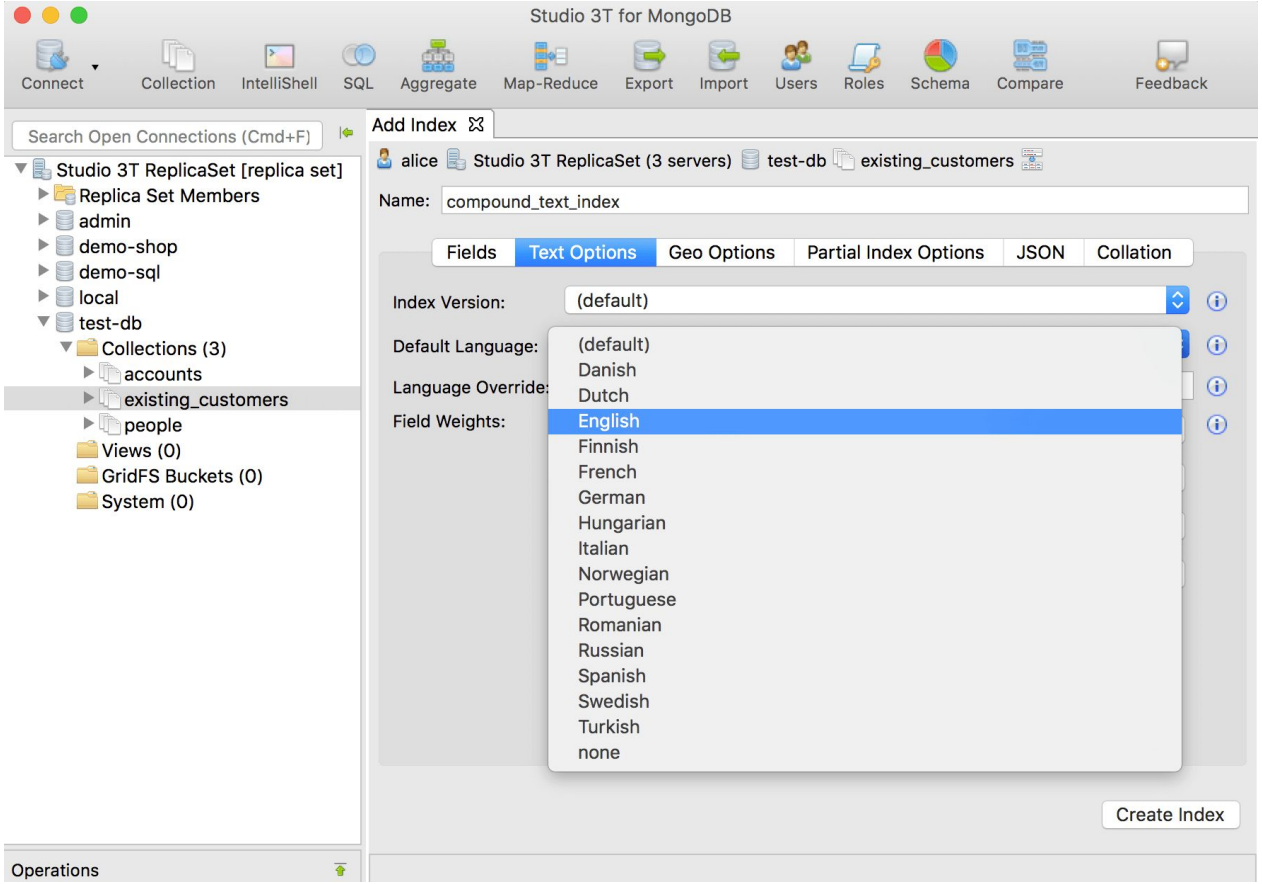
How to - Text indexing



How to - Text indexing



How to - Text indexing

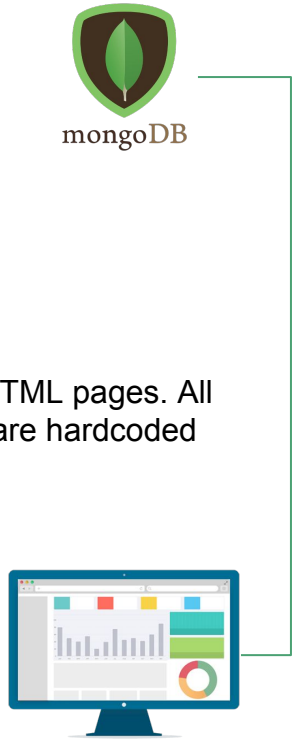


Querying with text indices

- db.stores.find({ $text: { $search: "java coffee shop" } })
  - $text tokenizes the search string and performs a logical OR
  - Will search on all indexed fields
  - Results include a relevance score for each record
- db.stores.find({ $text: { $search: "java \"coffee shop\"" } })
  - Will match exact phrases “java” OR “coffee shop”
-

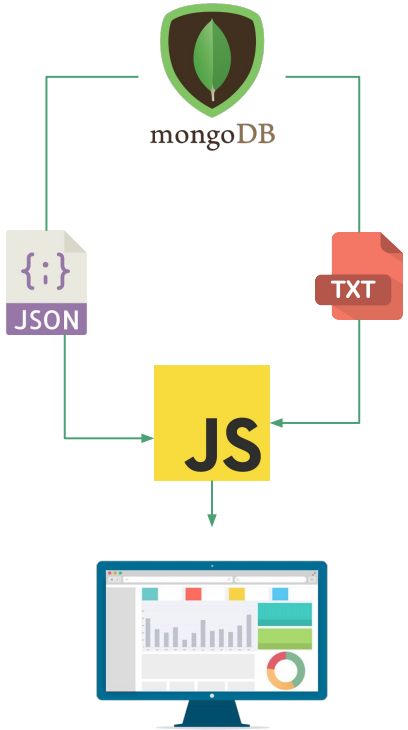
Web app development

Possibility #1

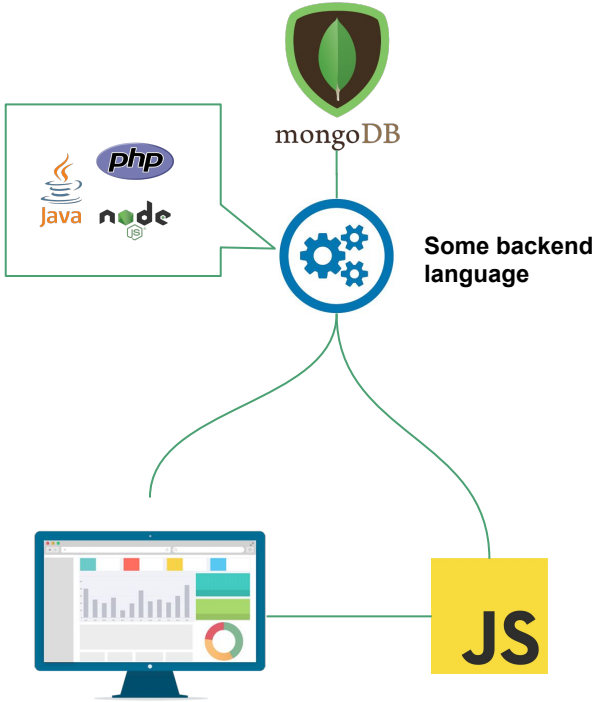


Static HTML pages. All results are hardcoded

Possibility #2



Possibility #3



A python-based microframework, suitable for small-scale applications.

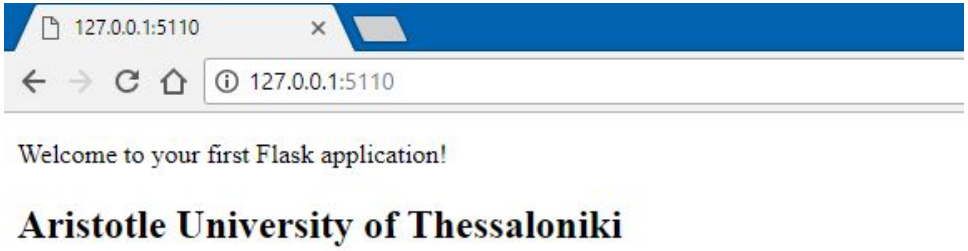
How to create a simple Flask App:

app.py

```python
from flask import Flask, render_template

app = Flask(__name__, template_folder='views')

@app.route("/")
def home():
    uni_name = "Aristotle University of Thessaloniki"
    return render_template('some_html_file.html', uni_name=uni_name)

if __name__ == "__main__":
    app.run(debug=True, host='127.0.0.1', port=5110)
```

some_html_file.html

```html
<!doctype html>
<html>
    <head>
        <meta charset="utf-8">
    </head>
    <body>
        <p>Welcome to your first Flask application!</p>
        <!-- Variable is passed, using the jinja2 templating engine -->
        <h2>{{ uni_name }}</h2>
    </body>
</html>
```



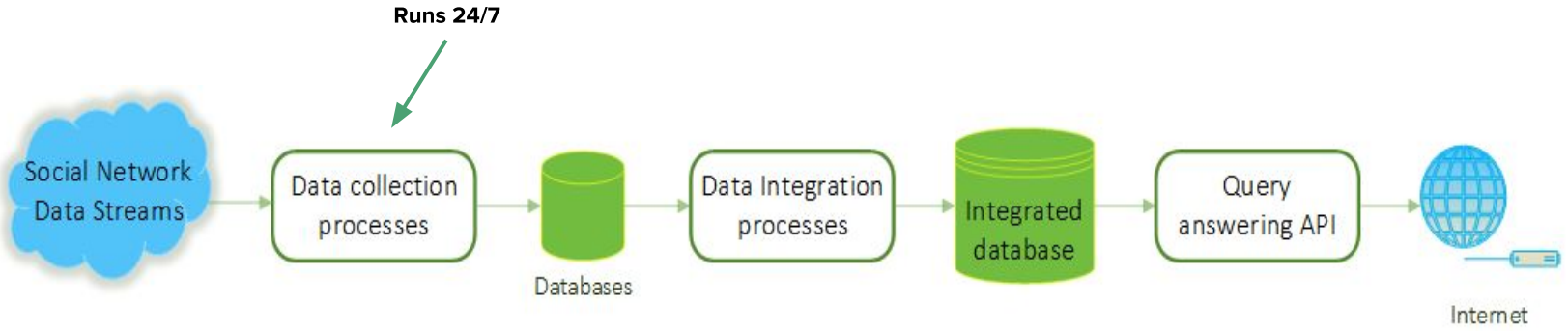
Diligent - A Social Media Data Integration platform

- Social media data explosion - Information Era
  - All data and metadata are saved into databases
  - Including private information (fingerprint, face, political views, etc.)

- Various data sources can be Integrated
  - Extract valuable insights
  - Personalized advertising
  - Sentiment analysis
  - -> Build a complete image - Increase data value $$

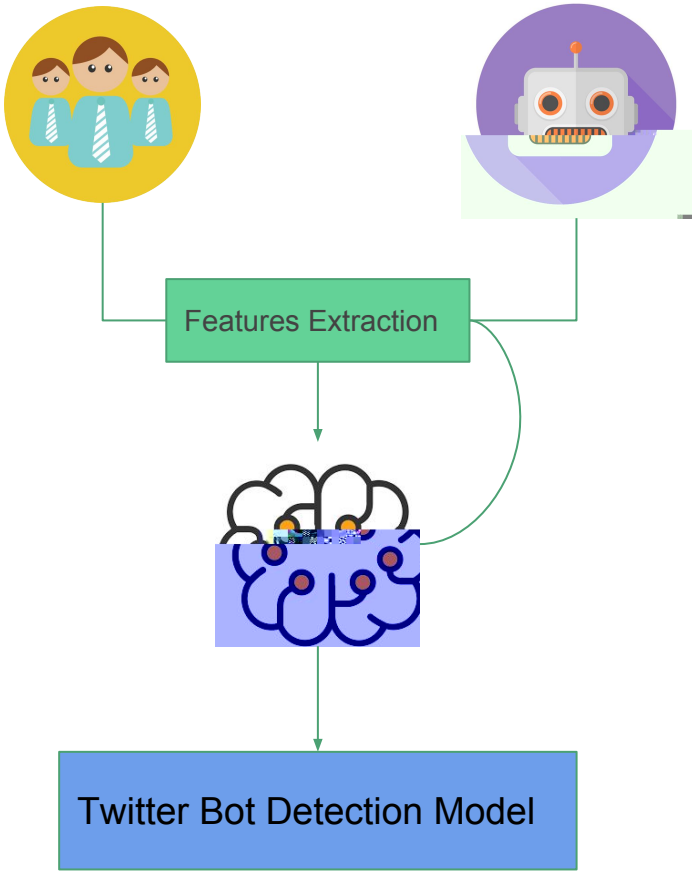
- Machine Learning opportunities
  - Integrated data as input

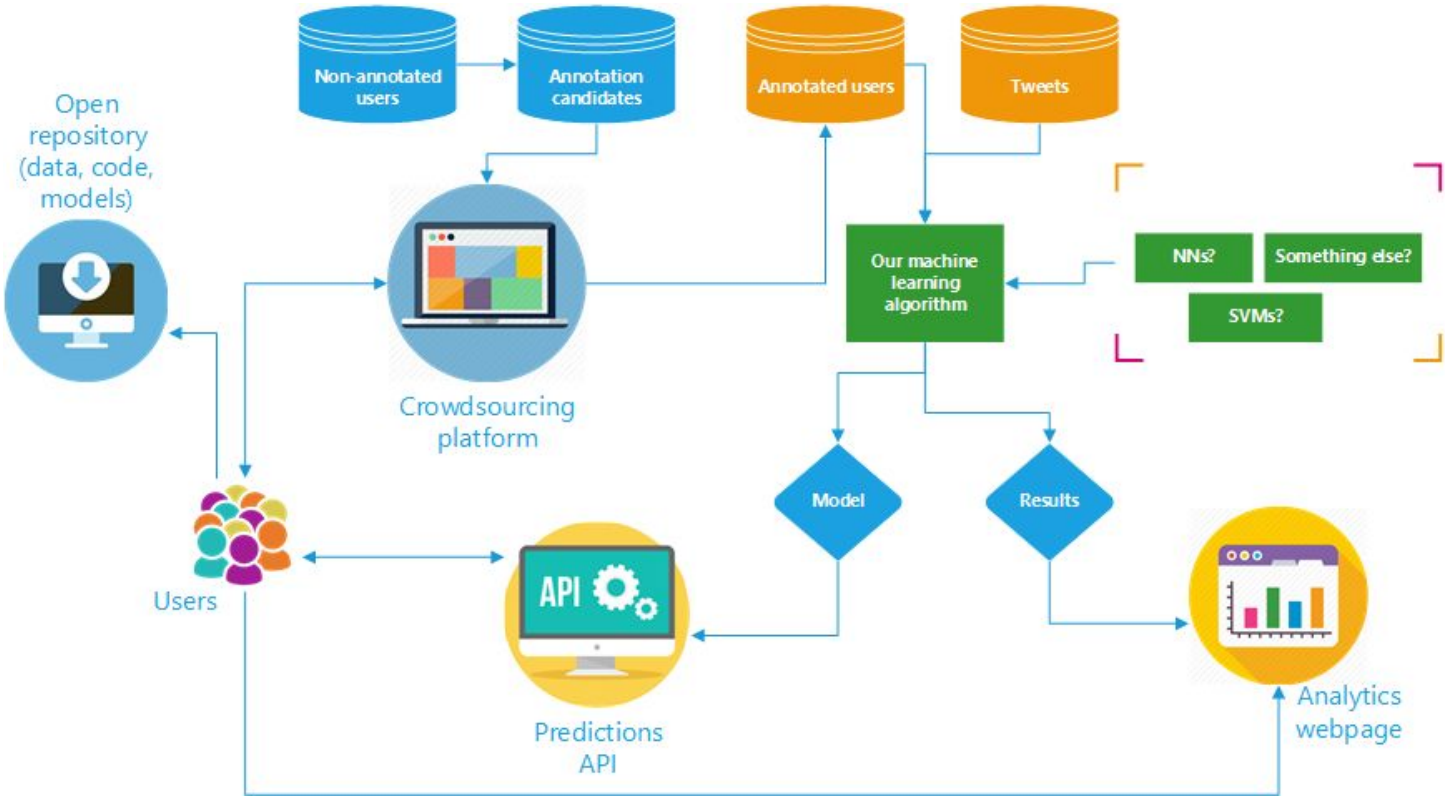
Diligent - System pipeline



Twitter bots detection

- Own a set of pre-annotated users as bots or humans
- Extract features from each user (focus on sentiment)
  - E.g. “num of tweets”, “tweets entropy”, “average sentiment polarity”
- Feed a machine learning algorithm all the features vectors
- Evaluate and extract model




Non-annotated users
Annotation candidates
Annotated users
Tweets
Open repository (data, code, models)
Crowdsourcing platform
Our machine learning algorithm
NNs?
Something else?
SVMs?
Model
Results
Users
API
Predictions API
Analytics webpage

Resources

- Dataset jsons
- Github link
- MongoDB
- Flask
- Flask-RESTful
- Python