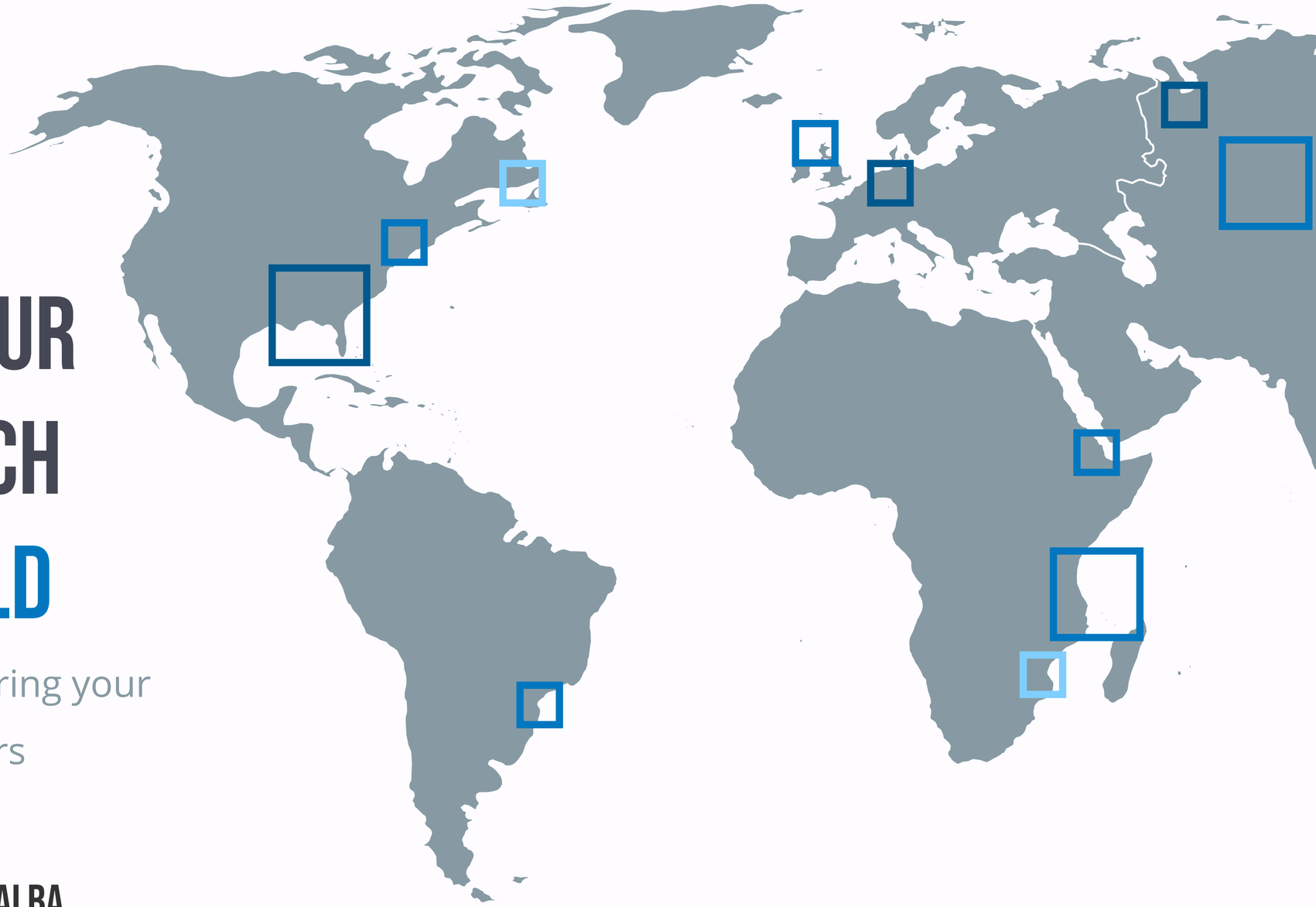


BRINGING YOUR NLP RESEARCH TO THE WORLD

Good practices on sharing your research with end-users

PRESENTED BY MARTÍN VILLALBA

PART 1

SHARING WITH FUTURE YOU

Good practices to keep everything neat and tidy

PART 2

SHARING WITH OTHER SCIENTISTS

Make other people want to use your code

PART 3

SHARING WITH THE WORLD

How to put your research on the web

GOOD PRACTICES ARE ALL ABOUT CLEAR COMMUNICATION

PART 1

SHARING WITH FUTURE YOU

PART 2

SHARING WITH OTHER SCIENTISTS

PART 3

SHARING WITH THE WORLD

SHARING WITH
FUTURE YOU

COMMENT YOUR CODE

SHARING WITH
FUTURE YOU

COMMENT YOUR CODE

```
def get_num_happy(text):
    return text.lower().count('happy')
```

SHARING WITH
FUTURE YOU

COMMENT YOUR CODE

- What does the function do?
- What are its parameters?
- What does it return?
- (Recommended) References

```
def get_num_happy(text):
    """ Returns how many times the word “happy” appears
    in a text. The search is case-insensitive.

    Parameters
    ----------
    text : str
        Un-normalized text that will be analyzed.

    Returns
    -------
    int
        How many times the word “happy” appears in the
        input text.

    Notes
    -----
    Algorithm taken from
    https://stackoverflow.com/questions/11800755
    """
    return text.lower().count(‘happy’)
```

SHARING WITH
FUTURE YOU

PUT YOUR CODE UNDER VERSION CONTROL

SHARING WITH
FUTURE YOU

PUT YOUR CODE UNDER VERSION CONTROL

- Learn: commit, pull, push, merge, branch
- Push your code to your favorite repo
- Nice to know: pull request
- Avoid: rebase, changing history
- Don’t panic: it’s not you, it’s them

```
def get_num_happy(text):
    return text.lower().count(‘happy’)
```

SHARING WITH
FUTURE YOU

PUT YOUR CODE UNDER VERSION CONTROL

- Learn: commit, pull, push, merge, branch
- Push your code to your favorite repo
- Nice to know: pull request
- Avoid: rebase, changing history
- Don’t panic: it’s not you, it’s them

```
def get_num_happy(text):
    return text.lower().count(‘happy’)
```

```
# git pull
# git add happy_function.py
# git commit -m ”Adds the first sketch of the sentiment analysis engine”
# git push
```

SHARING WITH
FUTURE YOU

PUT YOUR CODE UNDER VERSION CONTROL

- Learn: commit, pull, push, merge, branch
- Push your code to your favorite repo
- Nice to know: pull request
- Avoid: rebase, changing history
- Don’t panic: it’s not you, it’s them

```
def get_num_happy(text):
    return text.lower().count(‘happy’)
```

```
# git pull
# git add happy_function.py
# git commit -m ”Adds the first sketch of the sentiment analysis engine”
# git push
```

GitHub

GitLab

SHARING WITH
FUTURE YOU

ADD A README

readme.so

```
# Project Title
A brief description of what this project does and who
it's for.


## Installation
Instructions on how to install this project

```bash
  pip install my-project
  cd my-project
```

Usage/Examples
```bash
python happy_function.py
```

License
[MIT](https://choosealicense.com/licenses/mit/)
```

PART 1

SHARING WITH FUTURE YOU

PART 2

SHARING WITH OTHER SCIENTISTS

PART 3

SHARING WITH THE WORLD

SHARING WITH
OTHER SCIENTISTS

EASY: USE A VIRTUAL ENVIRONMENT

- Install libraries without calling your system administrator
- Isolates your dependencies
- Makes your environment reproducible

SHARING WITH
OTHER SCIENTISTS

EASY: USE A VIRTUAL ENVIRONMENT

- Install libraries without calling your system administrator
- Isolates your dependencies
- Makes your environment reproducible

```
# Install dependency
# pip install blingfire

# Save your dependencies
# pip freeze
blingfire==0.1.7
# pip freeze > requirements.txt

# Restore your dependencies
# pip install -r requirements.txt
```

SHARING WITH
OTHER SCIENTISTS

EASY: USE A VIRTUAL ENVIRONMENT

- Install libraries without calling your system administrator
- Isolates your dependencies
- Makes your environment reproducible

```
# Install dependency
# pip install blingfire

# Save your dependencies
# pip freeze
blingfire==0.1.7
# pip freeze > requirements.txt

# Restore your dependencies
# pip install -r requirements.txt
```

SHARING WITH
OTHER SCIENTISTS

MEDIUM: USE DOCKER

- Document every single step down to every file
- Forget about incompatible Operating Systems

- Resource heavy
- Commands are a little cryptic

SHARING WITH
OTHER SCIENTISTS

MEDIUM: USE DOCKER

- Document every single step down to every file
- Forget about incompatible Operating Systems

- Resource heavy
- Commands are a little cryptic

```
FROM python:3.7
# Set current directory in server
WORKDIR /my_app/

# Install dependencies
COPY requirements.txt ./
RUN pip install --no-cache-dir -r requirements.txt

# Copy source code and start server
COPY ./happy_function.py .
CMD [ "python", "happy_function.py" ]
```

SHARING WITH
OTHER SCIENTISTS

COMPLEX: PUBLISH A PACKAGE

- Official guide:
  HTTPS://PACKAGING.PYTHON.ORG/TUTORIALS/PACKAGING-PROJECTS/
- Poetry
  HTTPS://PYTHON-POETRY.ORG/
- Transformer?
  HTTPS://HUGGINGFACE.CO/

PART 1

SHARING WITH FUTURE YOU

PART 2

SHARING WITH OTHER SCIENTISTS

PART 3

SHARING WITH THE WORLD

SHARING WITH
THE WORLD

HAVE A SECURITY MINDSET

HTTPS://WWW.WIRED.COM/2008/03/SECURITYMATTERS-0320/

- Users will misuse your code. Be prepared.
- Always ask yourself “how could this go wrong?”

SHARING WITH
THE WORLD

HAVE A SECURITY MINDSET

HTTPS://WWW.WIRED.COM/2008/03/SECURITYMATTERS-0320/

```
def get_num_happy(text):
    return text.lower().count('happy')
```

- Users will misuse your code. Be prepared.
- Always ask yourself “how could this go wrong?”

SHARING WITH
THE WORLD

HAVE A SECURITY MINDSET

HTTPS://WWW.WIRED.COM/2008/03/SECURITYMATTERS-0320/

- Users will misuse your code. Be prepared.
- Always ask yourself “how could this go wrong?”

```
def get_num_happy(text):
    if isinstance(text, str):
        return text.lower().count(‘happy’)
    else:
        # The input is not a string. Therefore, the word
        # ‘happy’ never occurs
        return 0
```

SHARING WITH
THE WORLD

HAVE A SECURITY MINDSET

HTTPS://WWW.WIRED.COM/2008/03/SECURITYMATTERS-0320/

- Users will misuse your code. Be prepared.
- Always ask yourself “how could this go wrong?”

```
def get_num_happy(text):
    if isinstance(text, str):
        # Truncate the text to a maximum tweet length
        # because our model can’t deal with longer texts
        text = text[:280]
        return text.lower().count(‘happy’)
    else:
        # The input is not a string. Therefore, the word
        # ‘happy’ never occurs
        return 0
```

SHARING WITH
THE WORLD

ADD AN API

SHARING WITH
THE WORLD

ADD AN API

- Communicate with other computers over the Internet
- Programmers making use of your code will love it
- Deliver your results and let someone else take care of the interface

Flask

SHARING WITH
THE WORLD

ADD AN API

- Communicate with other computers over the Internet
- Programmers making use of your code will love it
- Deliver your results and let someone else take care of the interface


```
from flask import requests, jsonify

def get_num_happy(text):
    (…)

@app.route(‘/how_happy/’, methods=[‘POST’])
def endpoint_happy():
    query_text = request.args.post(‘query’)
    how_happy = get_num_happy(query_text)
    return jsonify({‘happy_times’: how_happy)
```

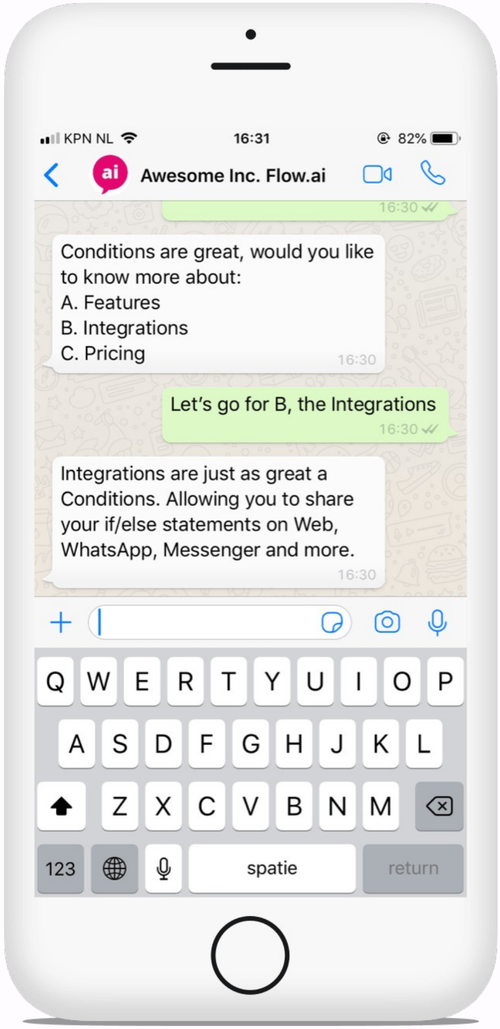
HTTP://WWW.YOURSERVER.COM/HOW_HAPPY/

SHARING WITH
THE WORLD

ADD AN API

- Communicate with other computers over the Internet
- Programmers making use of your code will love it
- Deliver your results and let someone else take care of the interface

Flask



© IMAGE: FLOW.AI

SHARING WITH
THE WORLD

BUILD A COMPLETE WEBSITE

SHARING WITH
THE WORLD

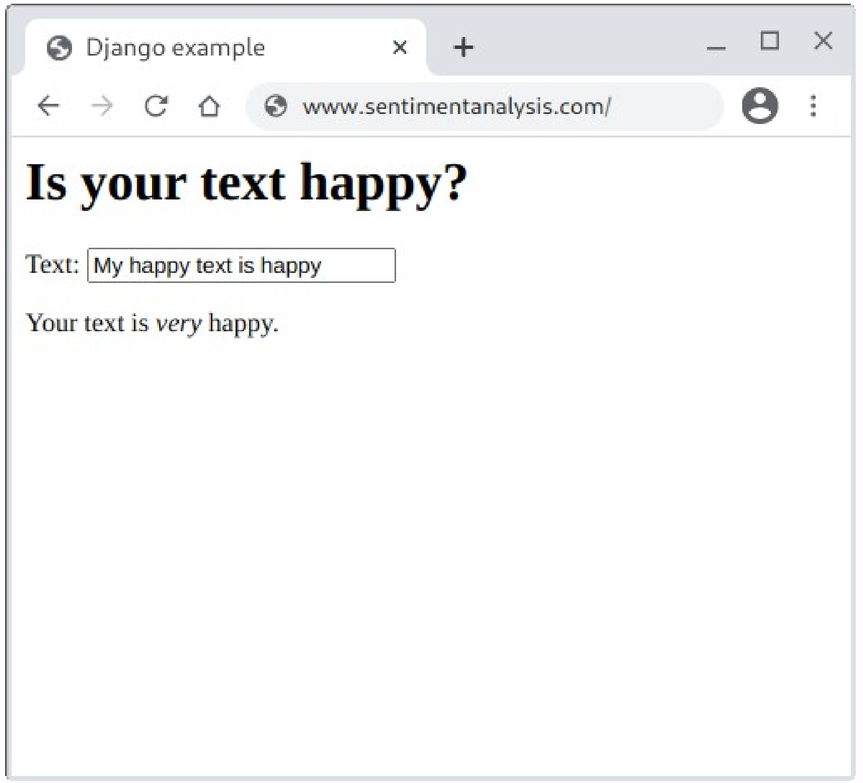
BUILD A COMPLETE WEBSITE

- Control the entire pipeline
- Plenty of templates and designs to get you started

SHARING WITH
THE WORLD

BUILD A COMPLETE WEBSITE

- Control the entire pipeline
- Plenty of templates and designs to get you started

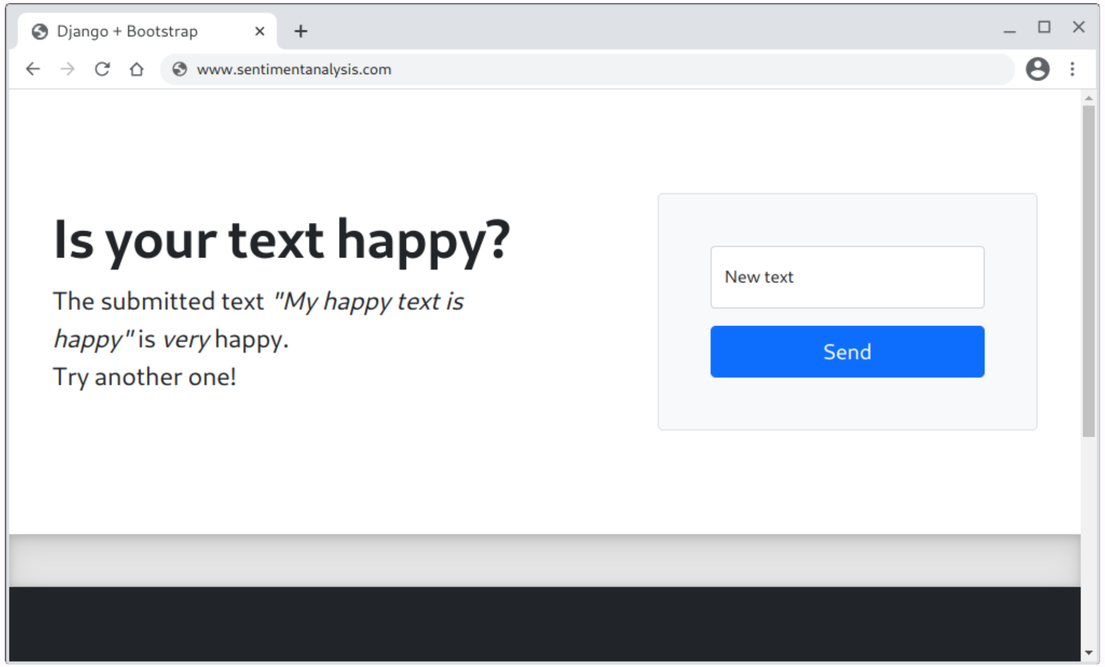


django

SHARING WITH
THE WORLD

BUILD A COMPLETE WEBSITE

- Control the entire pipeline
- Plenty of templates and designs to get you started
- Bootstrap: professional, responsive HTML ready to use



PART 1

SHARING WITH FUTURE YOU

Add comments, use version control, write a README.

PART 2

SHARING WITH OTHER SCIENTISTS

Make your environment reproducible.

PART 3

SHARING WITH THE WORLD

Build a website or create an API. Keep a security mindset!

THANK YOU FOR YOUR ATTENTION

PRESENTED BY

MARTÍN VILLALBA

VILLALBA@7COH.COM

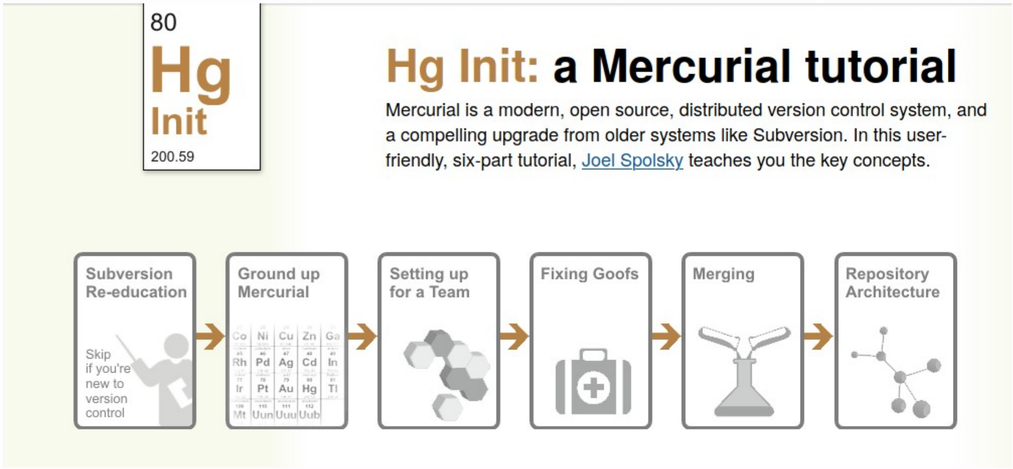
WWW.7COH.COM

EXTRA SLIDES

LEARN GIT THE
RIGHT WAY

MERCURIAL: GIT FOR HUMAN BEINGS

- Mercurial: similar to Git, but simpler
- Lost the Version Control Wars in 2019



HTTPS://HGINIT.GITHUB.IO/

SECURITY MINDSET

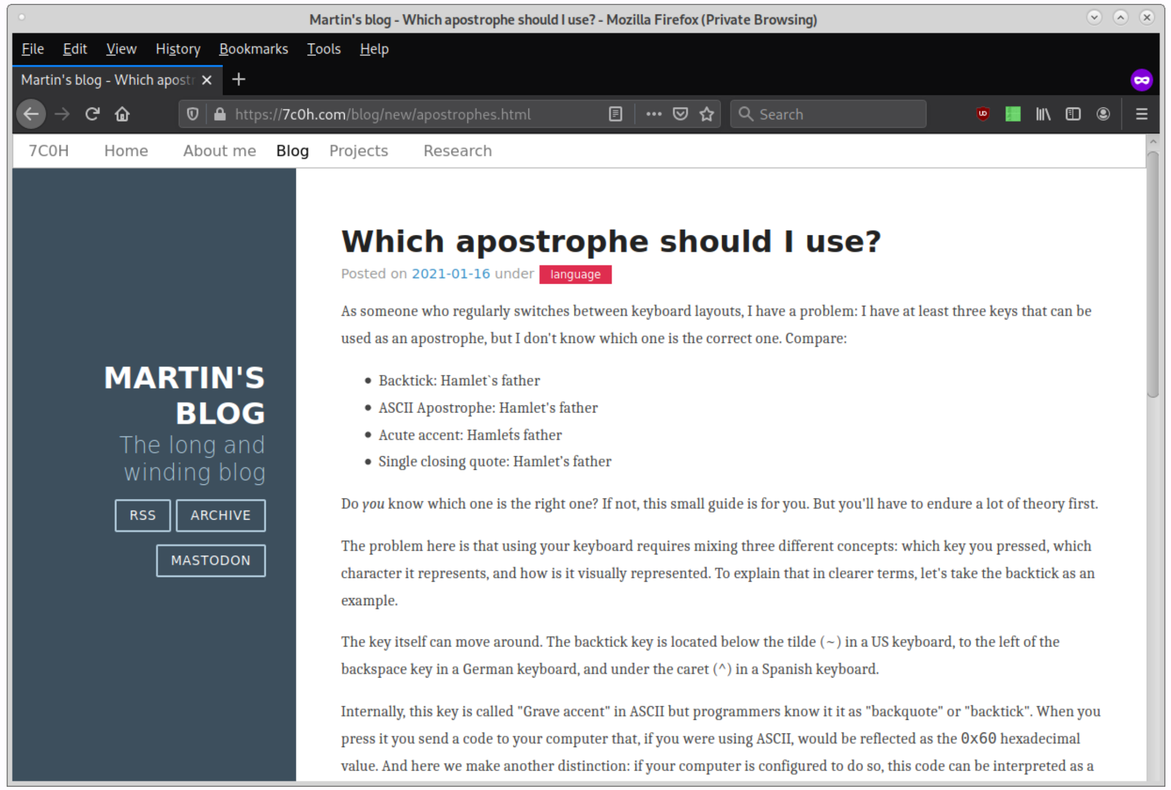
CROWDSOURCING: WHAT COULD GO WRONG?

- Multiple workers, same IP
- Same worker, multiple simultaneous IPs
- Multiple IPs from a country that doesn't speak your requested language
- Same control words across tasks
- Worker completes the task impossibly fast
- Worker uploads illegal material
- Workers access information given by other workers
- There are more workers asking for compensation than data in your database
- There are more records in your database than there are workers

REACHING OUT TO PEOPLE

HAVE A WEBSITE, WRITE A BLOG

- Website: make it easy for other to know everything about your research
- Keep a blog
  - Sharpen your writing skills
  - Talk about what you do in a clear language



INDUSTRY TRACK

HTTPS://2021.NAACL.ORG/CONFERENCE-PROGRAM/INDUSTRY/PROGRAM.HTML