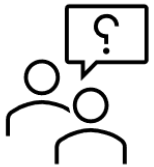
Crisis Events Webpages Archiving

Jeff Chen, Matthew Averyt, Nihal Mitta, Merna Khamis, Justin Lee
CS 4624: Multimedia, Hypertext, and Information Access

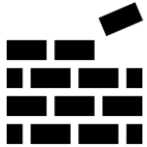
Mohamed Farag
Virginia Tech, Blacksburg VA 24061
4/20/23

Outline



Summary

Installation



Progress

What’s Next

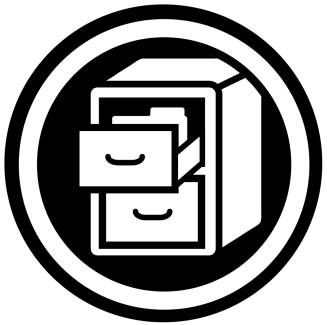
TimeLine

Acknowledgement

References

Summary

- Web pages disappear quickly, so it is important to archive them
- To help with archival, we created an application to record/replay web pages
  - Python/Flask backend
  - React frontend
- Current solutions aren't fit for mass archival and can be cumbersome to use



Installation

Software:

- Python 3.9.X
- Node.js
- npm
- Git bash
- Windows

Installation Instructions:

1. Git clone https://git.cs.vt.edu/jklee00/webarchive.git
2. Open git bash and enter the following:
    a. pip install warcio
    b. pip install requests
    c. pip install psutil
    d. pip install pywb
    e. pip install requests
3. Navigate to “webarchive\Backend\src” and open git bash./

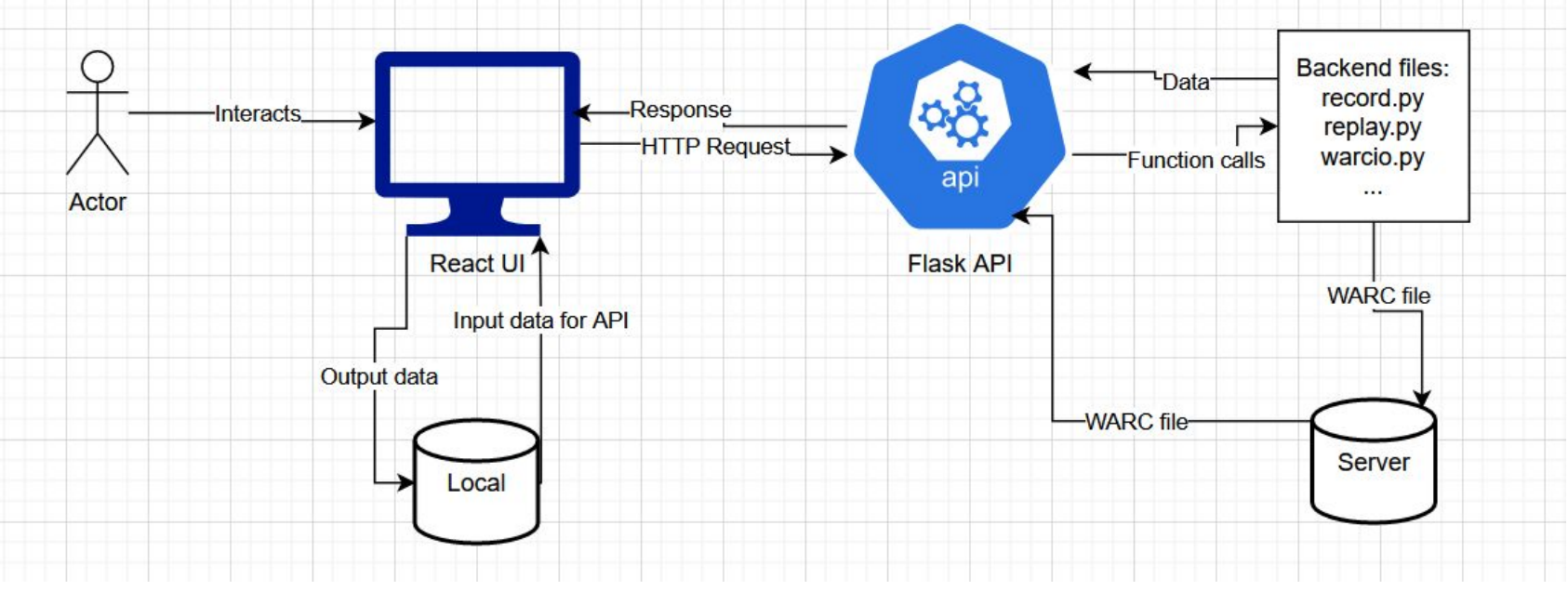
Installation

4. Enter into git bash:
    a. python -m venv venv
    b. "venv/Scripts/activate" with quotations.
    c. pip install flask
    d. pip install psutil
    e. pip install flask_cors
    f. pip install requests
    g. pip install warcio

To run the program, both Front-end and Back-end need to be run concurrently:

1. Front-end: navigate to “webarchive\Frontend\src”, open git bash, then enter “npm start” without quotations.
2. Back-end: navigate to “webarchive\Backend\src”, open git bash, enter “venv/Scripts/Activate” with quotations, then “python flask-app” without quotations.

System Overview



Current Progress - Recording

record.py 994 bytes — Open in Web IDE | Replace | Delete

```python
import subprocess
import os
import webbrowser
import time
import replay

def main():
    #root = os.path.dirname(os.path.dirname(os.path.realpath(__file__)))

    url = input("Input a url to record> ")
    collection_name = input("Input a collection name> ")

    os.chdir("../")
    # print(os.getcwd())

    # set up environment variables to make it work on windows
    env = os.environ
    subprocess.run([r"wb-manager", "init", collection_name], shell=True, env=env)
    # subprocess.run(["wb-manager", "add", collection_name])
    time.sleep(4)

    wayback = subprocess.Popen([r"wayback", "--record", "--live", "-a", "--auto-interval", "10"], shell=True, env=env)

    print("\n\nAfter a few seconds, recording will automatically halt")
    time.sleep(2)
    webbrowser.open_new_tab("http://localhost:8080/" + collection_name + "/record/" + url)

    time.sleep(20)

    wayback.kill()
    # replay.replay_collection()

if __name__ == "__main__":
    main()
    replay.replay_collection()
```

```
PS M:\Projects\webarchive\src> & "C:/Users/Jeff Chen/AppData/Local/Microsoft/WindowsApps/python3.10.exe" m:/Projects/webarchive/
src/record.py
Input a url to record> https://www.cbsnews.com/news/pennsylvania-chocolate-factory-explosion-rm-palmer/
Input a collection name> PennChocoFactoryExplosion
2023-03-25 14:01:43,375: [INFO]: Created Directory: M:\Projects\webarchive\collections\PennChocoFactoryExplosion
2023-03-25 14:01:43,376: [INFO]: Created Directory: M:\Projects\webarchive\collections\PennChocoFactoryExplosion\archive
2023-03-25 14:01:43,376: [INFO]: Created Directory: M:\Projects\webarchive\collections\PennChocoFactoryExplosion\indexes
2023-03-25 14:01:43,376: [INFO]: Created Directory: M:\Projects\webarchive\collections\PennChocoFactoryExplosion\static
2023-03-25 14:01:43,377: [INFO]: Created Directory: M:\Projects\webarchive\collections\PennChocoFactoryExplosion\templates
2023-03-25 14:01:43,377: [INFO]: Created Directory: M:\Projects\webarchive\static
2023-03-25 14:01:43,377: [INFO]: Created Directory: M:\Projects\webarchive\templates


After a few seconds, recording will automatically halt
2023-03-25 14:01:48,825: [INFO]: Auto-Indexing Enabled on "M:\Projects\webarchive\collections", checking every 10 secs
2023-03-25 14:01:48,934: [INFO]: Starting Gevent Server on 8080
2023-03-25 14:01:48,936: [INFO]: Checking Collection: chenjeff4840
2023-03-25 14:01:48,937: [INFO]: Checking Collection: example
2023-03-25 14:01:48,937: [INFO]: Checking Collection: PennChocoFactoryExplosion
2023-03-25 14:01:48,938: [INFO]: Checking Collection: PennChocolateFactoryExplosion
127.0.0.1 - - [2023-03-25 14:01:51] "GET /PennChocoFactoryExplosion/record/https://www.cbsnews.com/news/pennsylvania-chocolate-f
actory-explosion-rm-palmer/ HTTP/1.1" 200 5293 2.293716
127.0.0.1 - - [2023-03-25 14:01:51] "POST /live/resource/postreq?param.recorder.coll=PennChocoFactoryExplosion&url=https%3A%2F%2
Fwww.cbsnews.com%2Fnews%2Fpennsylvania-chocolate-factory-explosion-rm-palmer%2F&closest=now&matchType=exact HTTP/1.1" 200 36081
0.233222
127.0.0.1 - - [2023-03-25 14:01:51] "POST /live/resource/postreq?param.recorder.coll=PennChocoFactoryExplosion&url=https%3A%2F%2
```

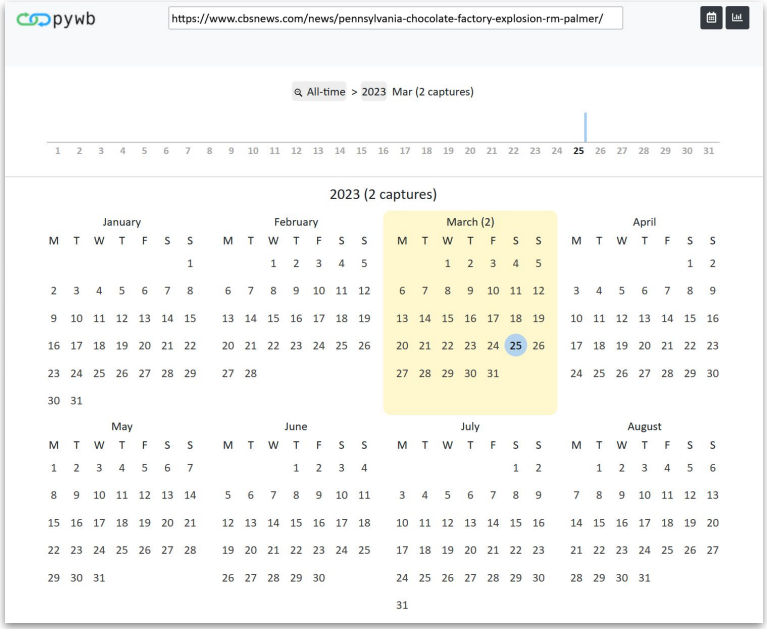
Current Progress - Multithreading

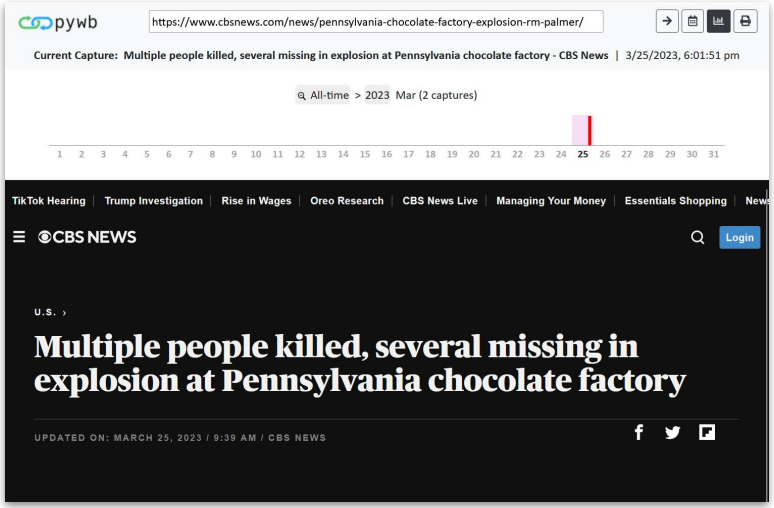
```
# Function added to thread. Retrieves URLS
def archive_urls(collection_name, url_list):
    # Using Requests, get URL info
    # Sleep

def main():

    # Set up URLs, paths, and Wayback machine
    # EXISTS ALREADY

    # Retrieve domains from URLs and sort into lists

    # For each list, create new thread using ThreadPoolExecutor
    # and call archive_urls()
    # Wait for threads to finish

    # Clean up Wayback Machine processes and directories
```

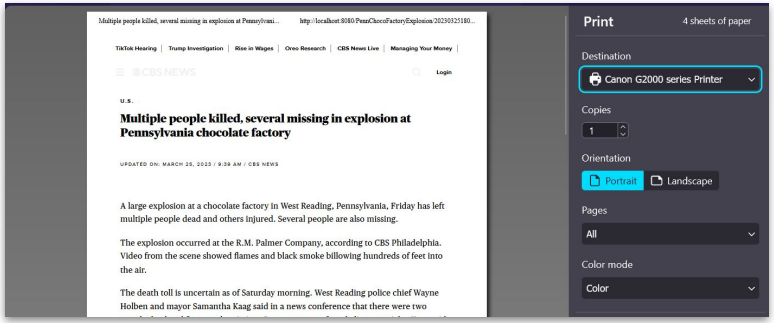
Current Progress - Full Backend

```python
def main(url_list):
    # Generate a unique collection name using the time stamp (using the day and time it was created)
    collection_name = "collection_" + datetime.datetime.now().strftime("%Y%m%d_%H%M%S")

    original_working_directory = os.getcwd()

    # Change the working directory to the location of this script
    script_directory = os.path.dirname(os.path.abspath(__file__))
    os.chdir(script_directory)

    # NOTE url_list is the path to the .txt file
    uploaded_file_path = url_list

    # Change directory to the top-level directory
    os.chdir(os.path.abspath(os.path.join(os.path.dirname(__file__), os.pardir)))

    # Initialize the collection
    subprocess.run(["wb-manager", "init", collection_name])
    time.sleep(4)

    # Set the Wayback to recording mode
    wayback = subprocess.Popen(["wayback", "--record", "--live", "-a", "--auto-interval", "10"])

    print("\n\nAfter a few seconds, recording will automatically halt")
    time.sleep(2)

    # iterate through the URLs from the uploaded file from the frontend and archive them
    with open(uploaded_file_path) as file:
        for line in file:
            url_to_archive = "http://localhost:8080/" + collection_name + "/record/" + line.strip() + "/"
            response = requests.get(url_to_archive)
            time.sleep(10)

    wayback.kill()

    # Change the directory to the original
    os.chdir(original_working_directory)

    # Build path to WARC file
    collection_dir = os.path.join(os.path.abspath(os.path.join(os.path.dirname(__file__), os.pardir)), "collections", collection_name)
    # NOTE warc file name method only works iff there is one WARC file in archive
    warc_file_name = os.listdir(os.path.join(collection_dir, "archive"))[0]
    warc_file_path = os.path.join(collection_dir, "archive", warc_file_name)
```

Current Progress - Replay







Current Progress - Replay

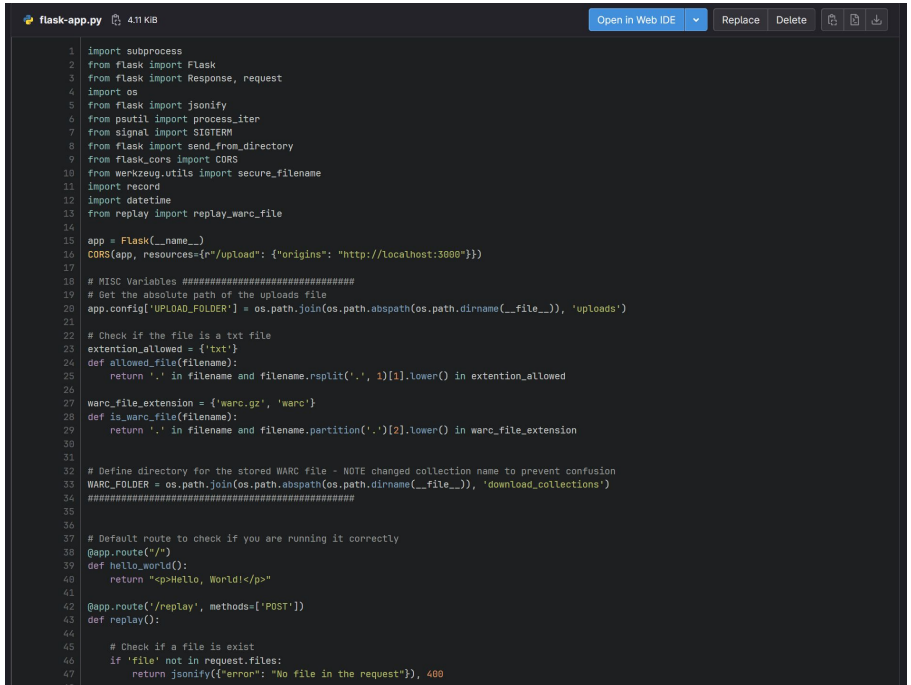
```
usage: warcioreader.py [-h] -w WARC [-v]

Get user input URLs from a WARC file.

options:
  -h, --help            show this help message and exit
  -w WARC, --warc WARC  WARC file to analyze
  -v, --verbose         Display statistical data, outputs url only by default

Author: Chenjeff4840 Last modified: 3/25/2023
```

warcioreader.py 3.04 KiB — Open in Web IDE | Replace | Delete

```python
import os
from multiprocessing.pool import ThreadPool
import time
import argparse
from warcio import ArchiveIterator

SPLITS = 5000

def main():
    """
    Given a WARC file, print all valid user defined URLs in the WARC file.
    """

    # Build commandline args
    parser = argparse.ArgumentParser(
                    prog='warcioreader.py',
                    description='Get user input URLs from a WARC file.',
                    epilog='Author: Chenjeff4840\t\tLast modified: 3/25/2023')

    parser.add_argument('-w', '--warc', required=True, help="WARC file to analyze", dest='warc')
    parser.add_argument('-v', '--verbose', help="Display statistical data, outputs url only by default",\
        action='store_true', dest='verbose')

    # Entrance pathway depending on parameters
    args = vars(parser.parse_args())

    # Get path
    warc_path = args["warc"]
    assert(os.path.exists(warc_path))

    start = time.time()

    # Get urls
    urls = set(get_url_list(warc_path=warc_path, verbose=args["verbose"]))

    # Print all urls
    for url in urls:
        print(url)

    if args["verbose"]:
        print("Runtime to process URLs only: {}s".format(time.time() - start))
        print("Number of URLS: {num}".format(num=len(urls)))

def get_url_from_records(records:list)->list[str]:
    """
```

```
PS M:\Projects\webarchive> & "C:/Users/Jeff Chen/AppData/Local/Microsoft/WindowsApps/python3.10.exe" m:/Projects/webarchive/src/warcioreader.py -w M:\Projects\webarchive\collections\chenjeff4840\archive\chenjeff4840.warc -v
Number of records found: 22706
https://sfstandard.com/research-data/california-mass-shooting-deaths-incidents-trend-data/
https://www.desertsun.com/picture-gallery/news/nation/2023/01/22/monterey-park-shooting-leaves-least-10-dead/11101572002/
https://www.sandiegouniontribune.com/news/california/story/2023-01-25/7-murder-counts-expected-in-half-moon-bay-farm-killings
https://theglobalherald.com/news/what-you-need-to-know-about-the-monterey-park-shooting-2/
https://www.sgvtribune.com/2023/01/24/monterey-park-massacre-puts-new-la-county-sheriff-on-global-stage/
https://www.foxla.com/news/lunar-new-year-shooting-vp-kamala-harris-to-travel-to-monterey-park-wednesday
https://www.denverpost.com/2023/01/25/shooting-monterey-park-des-moines-iowa-half-moon-bay/
https://www.cbsnews.com/sanfrancisco/news/half-moon-bay-mass-shooting-victims-identified-killed-with-purpose-with-intent-execution-style/
https://news.yahoo.com/victims-killed-monterey-park-shooting-165614491.html
https://napavalleyregister.com/news/state-and-regional/half-moon-bay-shooting-suspect-lived-at-mushroom-farm-where-4-were-killed-vice-mayor/article_1197772c-9c3c-11ed-8194-a38aaaf7fed6.html
https://www.usatoday.com/story/news/2023/01/25/hundreds-mourn-honor-victims-monterey-park-mass-shooting-attack/11117331002/
https://hildalsolis.org/remarks-of-supervisor-solis-as-prepared-for-delivery-on-moment-of-silence-for-monterey-park-mass-shooting-victims/
https://www.cbsnews.com/news/monterey-park-mass-shooting-los-angeles-california-deaths/
https://www.youtube.com/watch?v=N5WBPF6Vl6Q
https://www.post-gazette.com/news/crime-courts/2023/01/24/monterey-park-shooting-victims-motive-huu-can-tran/stories/202301240112
https://www.investing.com/news/world-news/suspect-in-californias-latest-shooting-rampage-due-in-court-2987595
https://www.npr.org/2023/01/24/1151184424/california-mass-shootings-latest-monterey-park-half-moon-bay
https://www.tribdem.com/news/monterey-park-shooting-victims-named-police-seek-motive/article_5bc1bfab-e4cc-5478-8aa2-3bba89402933.html
https://www.sfgate.com/politics/article/guns-from-recent-california-shootings-17738573.php
https://www.cnn.com/2023/01/25/us/california-three-mass-shootings-19-killed/index.html
https://abc7ny.com/harris-to-visit-monterey-park-to-meet-with-victims-families/12734896/
https://www.wbur.org/npr/1151409359/half-moon-bay-shooting-suspect-court
https://www.cnn.com/us/live-news/monterey-park-california-shooting-01-24-23/index.html
https://abcnews.go.com/US/video/monterey-park-shooting-96668176
https://jewishjournal.com/news/355446/latest-on-monterey-park-shooting-victims-identified-man-disarmed-gunman-in-alhambra/?utm_source=rss&utm_medium=rss&utm_campaign=latest-on-monterey-park-shooting-victims-identified-man-disarmed-gunman-in-alhambra
https://www.timesheraldonline.com/2023/01/24/locals-feeling-numb-from-recent-mass-shootings/
https://abc7.com/mass-shooting-monterey-park-gavin-newsom-half-moon-bay/12734894/
https://www.wsws.org/en/articles/2023/01/25/per1-j25.html
https://www.sgvtribune.com/2023/01/24/11-victims-killed-in-monterey-park-mass-shooting-now-identified/
https://mec.lacounty.gov/2023/press-releases/monterey-park-mass-shooting-victims-identified/
https://www.newindianexpress.com/world/2023/jan/26/happening-way-too-often-us-secret-service-report-delves-into-mass-attacks-2541365.html
Runtime to process URLs only: 3.020915985107422s
Number of URLS: 31
```

Current Progress - Flask Backend



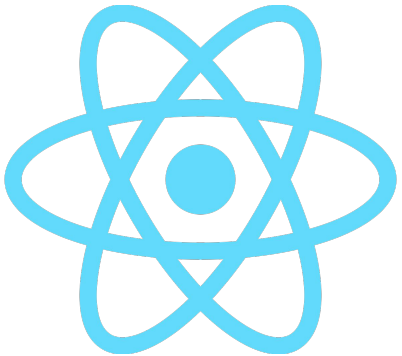
flask-app.py 4.11 KiB

Open in Web IDE | Replace | Delete

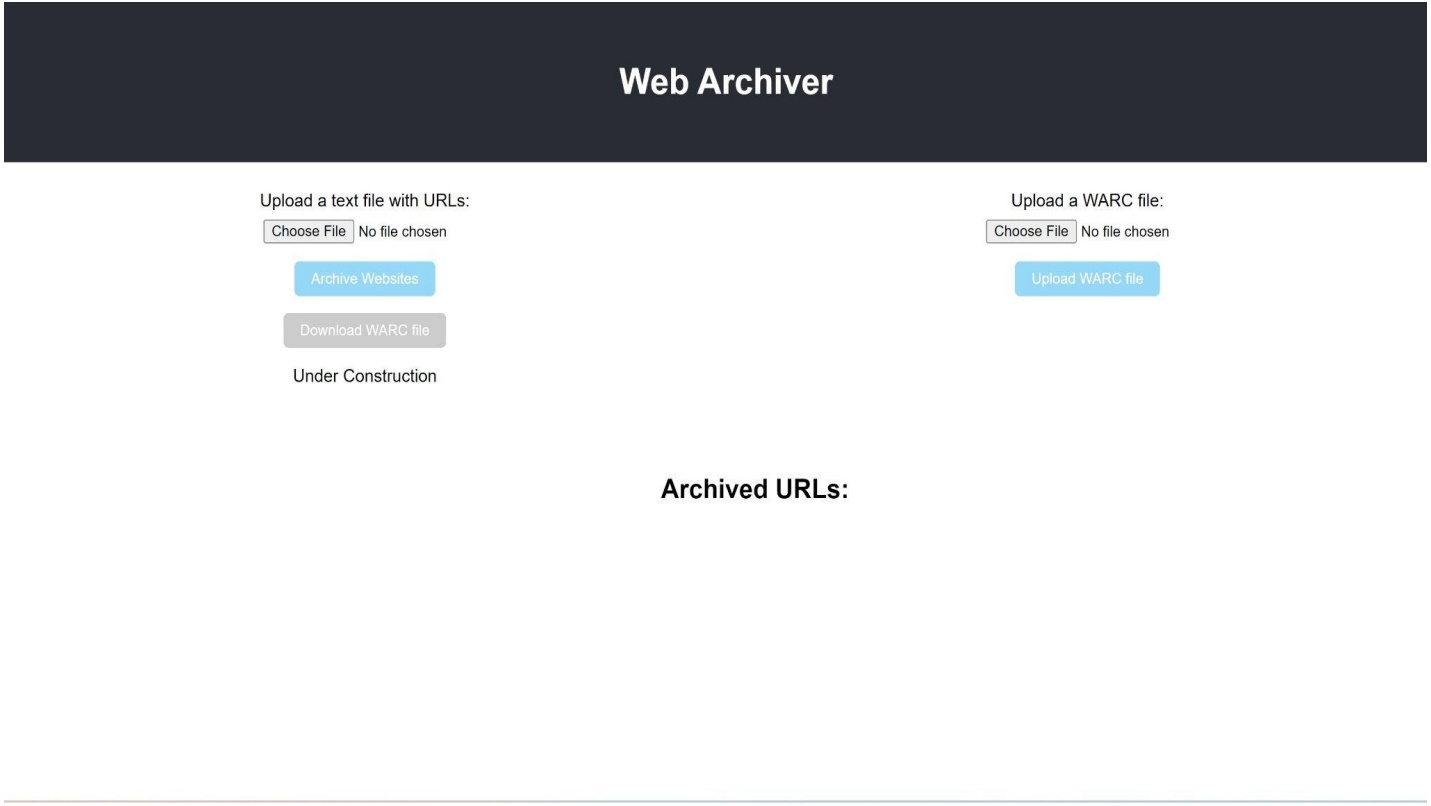
```python
import subprocess
from flask import Flask
from flask import Response, request
import os
from flask import jsonify
from psutil import process_iter
from signal import SIGTERM
from flask import send_from_directory
from flask_cors import CORS
from werkzeug.utils import secure_filename
import record
import datetime
from replay import replay_warc_file

app = Flask(__name__)
CORS(app, resources={r"/upload": {"origins": "http://localhost:3000"}})

# MISC Variables ##############################
# Get the absolute path of the uploads file
app.config['UPLOAD_FOLDER'] = os.path.join(os.path.abspath(os.path.dirname(__file__)), 'uploads')

# Check if the file is a txt file
extention_allowed = {'txt'}
def allowed_file(filename):
    return '.' in filename and filename.rsplit('.', 1)[1].lower() in extention_allowed

warc_file_extension = {'warc.gz', 'warc'}
def is_warc_file(filename):
    return '.' in filename and filename.partition('.')[2].lower() in warc_file_extension


# Define directory for the stored WARC file - NOTE changed collection name to prevent confusion
WARC_FOLDER = os.path.join(os.path.abspath(os.path.dirname(__file__)), 'download_collections')
###############################################


# Default route to check if you are running it correctly
@app.route("/")
def hello_world():
    return "<p>Hello, World!</p>"

@app.route('/replay', methods=['POST'])
def replay():

    # Check if a file is exist
    if 'file' not in request.files:
        return jsonify({"error": "No file in the request"}), 400
```

Work Done - Frontend

- React Web Application
  - Upload text file with URLs to archive.
  - Upload WARC file to replay.
- Connected to the backend using Flask.
- New UI with routes.
- Applied most of the new UI to our application.



Home Page


Web Archiver
Upload a text file with URLs:
Choose File No file chosen
Archive Websites
Download WARC file
Under Construction
Upload a WARC file:
Choose File No file chosen
Upload WARC file
Archived URLs:

Web Archiver

Archived URLs:

List of the successfully archived URLs

- https://www.washingtonpost.com/nation/2023/01/23/monterey-park-mass-shooting/
- https://abc7.com/monterey-park-shooting-victims-xiujuan-yu-kathleen-fong/12733402/
- https://www.nbcnews.com/news/us-news/monterey-park-police-search-motive-deadly-shooting-rcna66938
- https://www.wsj.com/video/watch-footage-shows-moment-man-disarms-monterey-park-mass-shooter/B2B981B0-1639-45AD-9281-703847972D4A.html
- https://www.nj.com/news/2023/01/nj-officials-issue-statements-on-monterey-park-mass-shooting.html
- https://www.independent.co.uk/news/world/americas/monterey-park-shooting-california-chinese-new-year-b2266860.html
- https://www.nbcnews.com/news/us-news/monterey-park-shooter-was-mad-world-not-just-ex-former-friends-say-rcna67331
- https://www.buzzfeednews.com/article/clarissajanlim/victims-monterey-park-shooting
- https://abcnews.go.com/US/multiple-dead-shooting-monterey-park-california/story?id=96592246
- https://6abc.com/mass-shooting-monterey-park-shooter-motive-huu-can-tran-lunar-new-year/12728098/
- https://time.com/6249372/monterey-park-shooting-victims/
- https://www.northjersey.com/story/news/2023/01/22/monterey-park-shooting-killing-10-new-jersey-asians-reel-as-lunar-new-year-begins/69830194007/

Web Archiver

Upload a text file with URLs:

Choose File No file chosen

Archive Websites

Download WARC file

Under Construction

Upload a WARC file:

Choose File WebArchiving.warc

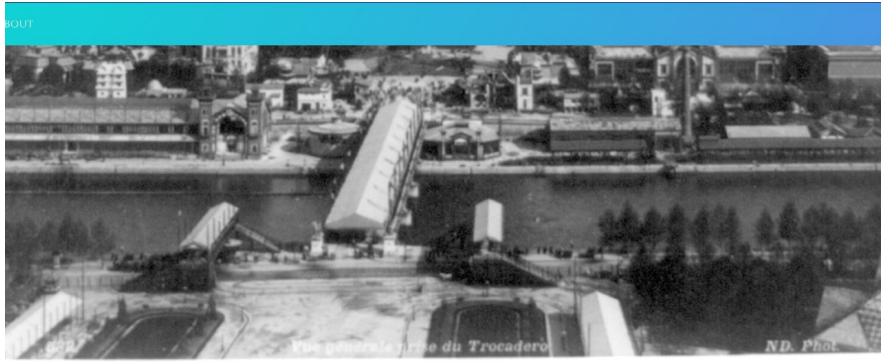
Upload WARC file

Selected WARC file: WebArchiving.warc

If the file is in the right format, its name will be displayed here instead of an error message

Archived URLs:

What's Done: Current UI



The Best Web Archiver You Ever Used

Say hello to something new the others haven't been able to do. The fastest, convenient, and most reliable web archiver in the world.

Multithreading

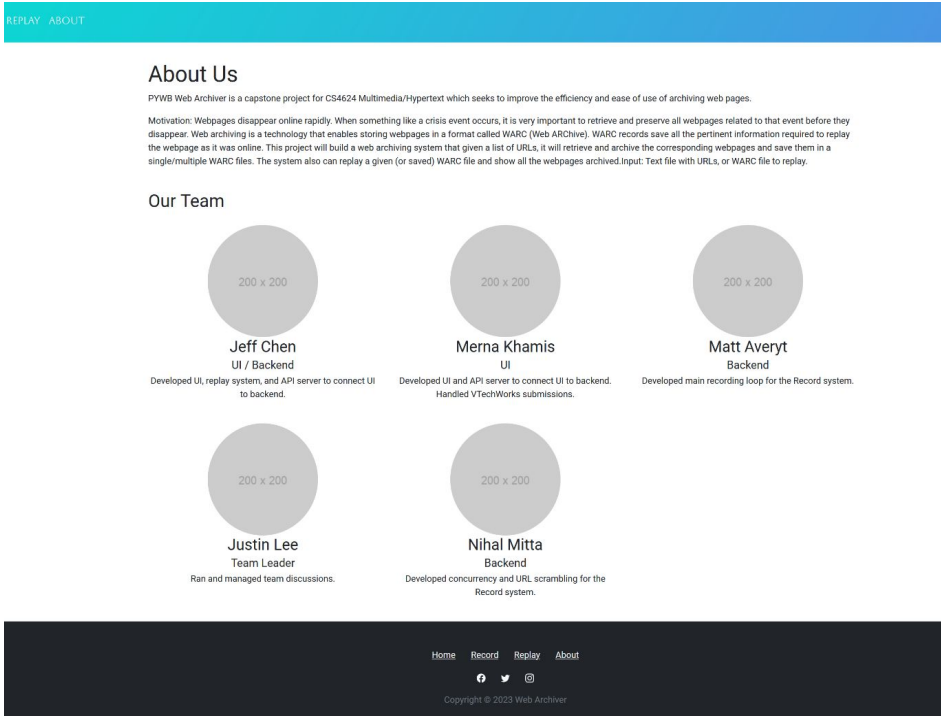
Tired of slow, web archivers being too faithful to the past? We are too so we made ours as fast as possible with multithreading. Yay 21st century!

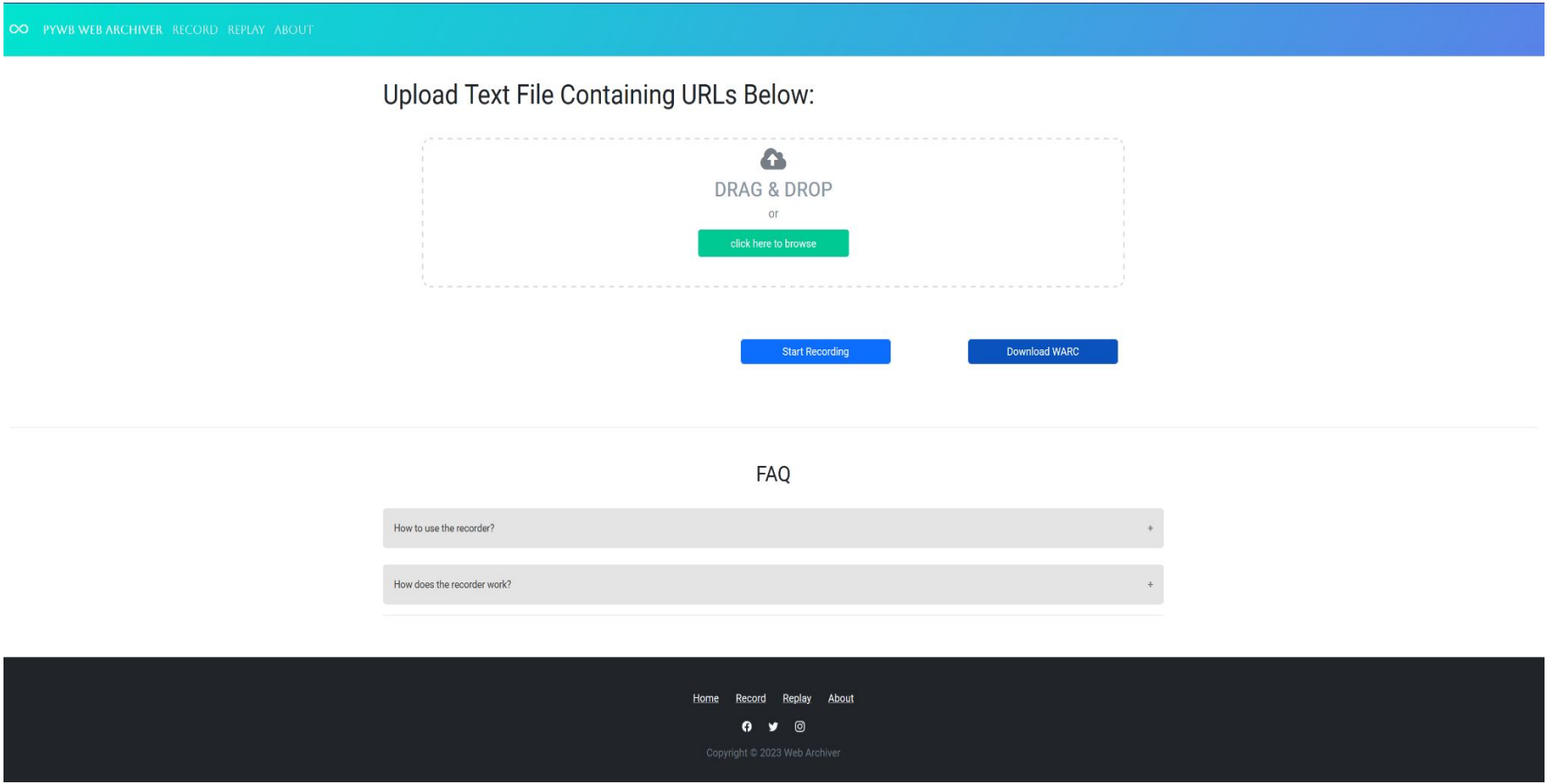
Bulk Archival

Sorry, you are too lazy to manually type in and wait for 100 urls to be archived!? No problem, you can give us all your urls at once and we'll archive them all for you with a touch of a button.

Local

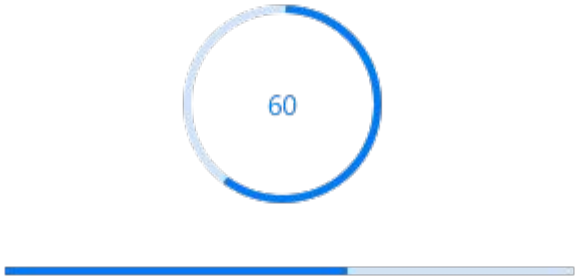
Everything from this website to the web recorder and web replay runs locally. 100% uptime and the best speeds you'll ever see from a web archiver.




PYWB WEB ARCHIVER RECORD REPLAY ABOUT
Upload Text File Containing URLs Below:
DRAG & DROP
or
click here to browse
Start Recording
Download WARC
FAQ
How to use the recorder?
How does the recorder work?
Home Record Replay About
Copyright © 2023 Web Archiver

What's Next

- Progress bar for the URL archival
  - Allows user to see current progress
- Path specification for archiving
- Mac and Linux Compatibility
- Quickness and efficiency
- Refine current UI
- Download WARC File after archiving is done
- List of URLs in WARC file instead of searching

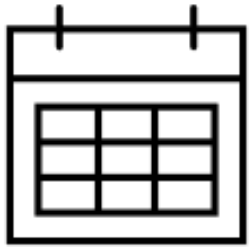


Lessons Learned

- Web archival
  - Importance and tools
- Flask
  - Frontend & Backend integration
- Python libraries
  - Pywb, Warcio
- Teamwork & Communication
  - Essential as a group
- Time management
  - Meet deadlines
- Meetings
  - Help get everyone up to speed

Timeline/Milestones

- Presentation 2: 3/28
- **All coding completed:** 4/9-16
- **User/Developer Manuals:** 4/9-16
- Final Presentation: 4/20
- Final Report: 4/27



Acknowledgement

Dr. Mohamed Farag

Research Associate of the Center of Sustainable Mobility VT, Transportation Institute.

We would thank our client, Dr. Farag, for allowing us to work on this project and working with us by giving us advice and guiding us through the entire process.

References

- [1] https://webrecorder.net/
- [2] https://www.selenium.dev/
- [3] https://pywb.readthedocs.io/en/latest/
- [4] https://github.com/internetarchive/heritrix3
- [5] https://github.com/webrecorder/browsertrix-crawler
- [6] https://legacy.reactjs.org/docs/getting-started.html

Thank you!

DEMO TIME