Automated Tool For Crisis Events
Collection Building

Brian Hays, Mitchel Rifae, Trevor Kappauf, Alex Zhang, Parsa Nikpour

Course: 83400 Multimedia/Hypertext
Instructor: Mohamed Farag
Virginia Tech, Blacksburg VA 24061, 11/16/2023
Outline

● Client and the Problem
● Project Overview
● Deliverables
● Features List
● System Architecture
● DEMO
● Work Completed
● Future work
● Acknowledgements
● References
Client and Problem

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

Our client needs an application that creates a collection of web pages that are related to a specified crisis event.
Problem Overview

● Build a web interface for crawling web pages about a given crisis event

● User input will be in the form of a list of new line delimited URLs

● Crawler software will gather outgoing links from inputted URLs and repeat the crawling process

● Final output will be a collection of derived webpages
Deliverables

- A Web Application capable of running such crawls, accompanied by it’s source code.
- User manuals of the web interface and the web crawler
- Developer manuals of the web interface and the web crawler
Features List

- User customized run
- Run crawl on input file
- Descriptive statistics of crawls
- Export pages and statistics
- Filter by specific crisis events
- User authentication
How the Crawler Works

1. Input text is parsed
2. Save URLs in Max Heap
3. Crawl and find more URLs
4. Pop URL
5. Gather statistics
6. Push information to DB
DEMO
**Work Completed**

**Preliminary Meetings**
Meet with client to discuss relevant details and constraints with project.

**Mockups**
Create Figma mockups for frontend and create backend structure.

**Front End Prototype**
Web app is set up and functional with minimal back end.

**Web Crawler Finalized**
Implement web crawler software into prototype web application.

**Front End Statistics**
Front end displays descriptive statistics correctly.

**Final Testing / Finish Touches**
Testing and bug fixing finished as well as source code and documentation.

**Timeline**
- 09.13: Preliminary Meetings
- 09.20: Mockups
- 10.04: Front End Prototype
- 10.18: Web Crawler Finalized
- 11.10: Front End Statistics
- 11.12: Final Testing / Finish Touches
Future Work

- Hosting to Linux Server
- Multithreading
- More statistics
- Greater user customization
Acknowledgements

Dr. Mohamed Farag

We would like to thank our client, Dr. Farag, for meeting with us weekly to help us understand the project scope and giving us implementation advice.
References

https://firebase.google.com/docs
https://chat.openai.com/
https://flask.palletsprojects.com/en/3.0.x/