PandemicTracker

A project by David, Mar, Daniel and Máté

Problem:

No platform provides aggregate information on:

- 1. Country preparedness
- 2. Travel bans

- :. High health cost for the government during pandemic
- ∴ People can be more aware and <u>avoid infection</u>

Solution:

Features & Functionalities:

- 1. Data retrieval
- 2. Data storage
- 3. Report & Dashboard
 - a. Global and national
- 4. Country comparisons

DEMO

Process Model - Waterfall Model

- Requirements well understood: Our product is not being ordered by any external customers, but rather we are building it ourselves and offering it governments
 - We have complete control over requirements.
- **Development is linear**: Our development followed a linear flow that begins with communication of requirements and ends with deployment.
 - There were no increments or expansions after the requirements were established.
- Product is not growing or changing: This product does not save any user
 information. Instead, its sole purpose is to scrape, organize, and display data.

Division of Labor - Gantt Chart

PandemicTracker Team	Gantt	Char	t											Proje	ect Du	ie: 14th	May 2	2020										
TASK NAME	START DATE	END DATE	DURATION (WORK DAYS)	TEAM MEMBER	PERCENT COMPLETE	M	 	th Apr.	-	M		th Ap			/EEK 3	3: 26-3 W		r. F		WEEK T		h May	F	M	EEK 5:	10-14 W		y F
Researching and Designing Data Retrieval System						IVI	VV	In	r	IVI	VV	III	Г	IVI	•	VV	In	Г	IVI	1	VV	In	г	IVI	1	VV	Th	Ĺ
Compile List of Key Websites	4/13	4/14	2	Máté	100%																							
Research API requests	4/14	4/16		Maria	100%																							
Construct Data Compilation Plan	4/16	4/22		Maria	100%																							
Write Beautiful Soup Python Script	1/14	1/19	1	Maria	100%																							
Test Script and Data Validity	1/17	1/21		Máté	100%																							
Researching and Designing Data Storage System																												
Research Data Storage Techniques	2/5	2/8	4	Maria	100%																							
Compile Overarching Data Storage Plan	2/11	2/16		Máté	100%																							
Write Data Scrapers for Health, Cases and Travel	2/11	2/16	-	Maria	100%																							
Restrictions Data	2/11	2/16		Máté	100%																							
Connect Retrieval System To Data Storage System	2/14	2/23		Maria	100%																							
Building Front End User Interface																												
Design an Outline for Website User Interface	3/5	3/8	4	Daniel	100%																							
Research Website Hosting Methods	3/8	3/13		Sangjin	100%																							
Research Website Creating Techniques	3/11	3/16		Sangjin	100%																							
	3/14	3/19		Sangjin	100%																							
Construct Website User Interface (front-end)	3/17	3/21		Daniel	100%																							
Evaluating Project Outcome																												
Test Website Design and UI	4/5	4/8	2	Daniel	100%																							
Test Data Rendering and Validity	4/8	4/10		Sangjin	100%																							
Survery Users to Collect Feedback	4/8	4/10		Maria	100%																							
Test Scrapers and Database	4/11	4/16	0.000	Máté	100%																							

Implementation - Frontend

В

- Templates written in HTML
 - Javascript and CSS used in conjunction with Flask Jinja
- 5 main templates:
 - o Home Page
 - Country List
 - Compare Countries
 - Single Country
 - Health Page
- Bootstrap used to polish interface





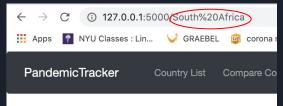
Implementation - Frontend



1

2.







Implementation - Frontend



3.

PandemicTracker	Country List	Compare Cour	ntries Hea	Ith Guidelines	About			
					Count	ry List	t	
		Country Name	Num Cases	Num Deaths	Num Recovered	Doctors /1000 People	Hospital Beds /1000 People	Latest Travel Restriction
	8	Afghanistan	4402	120	558	0.28	0.5	None
	樂	Albania	868	31	650	1.2	2.9	Closed borders on 15 March and all commercial flights on 29 March. Therefore, travel between Albania and Montenegro, Kosovo, North Macedonia, and Greece is prohibited, unless for the transportation of goods.
	C	Algeria	5723	502	2678	1.83	1.9	All international flights to Algeria are suspended.
	(B)	Andorra	755	48	550	3.33	2.5	None
	এ	Angola	45	2	13	0.21	0.8	None
Showing 1 to 185 of 185 entries								

4.



Implementation - Back-end

- Framework: Flask
- Database: MySQL
- Flask + MySQL = Pymysql library
- Scraping: Python BeautifulSoup and Pandas







Quality Assurance and Testing

- Each team member wrote test cases for the parts of the program that they wrote, to ensure that they could identify all possible defects.
- We tried our best to be cautious when fixing bugs: asking what other bugs a
 fix could introduce and doing regression testing after major fixes.
- Overall, we...
 - Made 13 test cases
 - o Identified (and fixed!) 4 major bugs through those test cases
- The end result is a **robust backend** that can respond to and handle changing external circumstances, and a **responsive frontend** that seamlessly draws data from it.

Future Maintenance

1. Maintenance

- a. PandemicTracker is **entirely dynamic**: as long as the data sources it relies on remain updated, it will present up-to-date data without any need for manual updating
- b. The data sources will need to be periodically checked for updatedness and accuracy

2. Future features

- a. Live deployment!
- b. Making scraping faster/more efficient
- c. Integration of more information