# IEPS Poročilo 1 - Ekipa “Vipavska Burja”

Matjaž Rupnik, Blaž Ličen

Marec 2021

# 1 Implementacija

Pajka smo spisali tako, da uporablja podatkovno bazo (v nadaljevanju PB) kot repozitorij za shranjevanje podatkov o spletnih straneh, hkrati pa jo uporablja tudi kot “frontier”. To deluje na način, da so strani, ki so v frontierju, v tabeli “page”, v stolpcu “page_type_code” označene kot “FRONTIER”. Ob zapisu v frontier se novi povezavi samodejno določi naslednji “id”, kar skrbi, da se povezave iz frontierja jemljejo v pravilnem vrstnem redu oz. po načinu BFS.
Ob zagonu pajka se preveri ali je frontier (v PB) prazen. Če je prazen to pomeni, da je to prvi zagon pajka, zato se sproži funkcija “initFrontier”, ki napolne frontier s podanimi URL naslovi. Če pa je v frontierju več kot 1 zapis tipa “FRONTIER” pomeni, da je pajek na znova zagnan, zato se zažene funkcija “initFrontierProcessing”, ki spremeni vse zapise tipa “PROCESSING”, v tip “FRONTIER”. To je varovalka, če se pajek ustavi sredi procesiranja strani in mu ne uspe zapisati vseh podatkov o strani v bazo, se bo pajek ob ponovnem zagonu ponovno obiskal stran. Nato se v ukazni vrstici uporabnika vpraša, s koliko procesi naj pajek deluje. Zatem se zažene izbrano število procesov.
Ker je bila baza prevelika za oddajo na GitHub, smo jo naložili na drive in je dostopna na povezavi: https://drive.google.com /drive/folders/1062HdYvG427uvF1cgZTS973DzcBEowlS?usp=sharing

## 1.1 Delovanje funkcije process

Vsak proces predstavlja zagon funkcije “process”. Ta funkcija je ubistvu velika zanka, ki na začetku dobi naslednji URLiz baze (to stori tako, da pokliče funkcijo “getNextUrl”), nato preveri ali je domena te strani že v tabeli “site”, če ni se na domeni poskusi dostopati do robots.txt. Če pravila obstajajo se shranijo v PB. Če pa je domena že v PB, iz baze naloži pravila za robots, če obstajajo. Nato se preveri ali sploh lahko gremo na ta URL. Če ne smemo iti na url se v PB zapiše “NOTALOWED”, če pa lahko gremo na ta URL se najprej iz samega URLja preveri ali gre za “BINARY” datoteko. Če ja, potem se v bazo zapiše da povezava kaže na “BINARY”, drugače pa se sproži funkcija “fetchPageContent”, ki pridobi HTML vsebino iz URL povezave. Na tem mestu se izračuna hash vsebine, če je vsebina podvojena se v podatkovno bazo zapiše “DUPLICATE”, če pa je vsebina nova oz. nikoli videna, se nato pokliče funkcija “getHrefUrls”, ki iz vsebine pridobi vse “<a href=” povezave in jih zapiše v frontier. Nato se pokliče funkcija “getJsUrls”, ki iz vsebine pridobi vse “onclick” povezave in jih zapiše v frontier. Nato se pokliče funkcija “getImgUrls”, ki iz vsebine pridobi vse “<img src=” povezave in jih zapiše v PB v tabelo “image”. Kanonizacija url se izvaja ob zapisu v podatkovno bazo, za to nalogo se uporablja knjižnjica url-py.

Pajek uporablja knjižnico Selenium-Wire, za prenos spletne strani. To knjižnico smo izbrali namesto navadne Selenium, ker ponuja na ogled tudi vse odgovore strežnika,. Tako smo dobili HTTP odzivno kodo in tip strani iz odgovora strežnika.

## 1.2 Delovanje funkcije getNextUrl

Implementirali smo dva načina delovanja funkcije getNextUrl:

Delovanje BFS (implementirano v datoteki fri-wier-vipavska-burja_BFS.py):
Ob prvem zagonu funkcija dobi parameter firstSleep. To je rešitev problema prvega zagona, ko so vsi procesi na enkrat klicali funkcijo in vsi hkrati dobili isti URL iz frontierja. Tako pa vsak proces pokliče funkcijo z daljšim časovnim zamikom, zato da ima funkcija čas dodeliti vsakemu procesu svoj URL iz frontierja. Funkcija deluje na enostaven način. Iz PB frontierja pridobi url z najmanjšim id, ter označi, da je ta URL trenutno v obdelavi (v stolpcu “page_type_code” nastavi vrednost na “PROCESSING”)

Delovanje po načinu vsak proces dobi svojo domeno (implementirano v datoteki fri-wier-vipavska-burja.py):
Delovanje pajka smo pohitrili z drugačno implementacijo funkcije getNextUrl, ki deluje na način, da vsakemu procesu dodeli nov URL, kateri pripada domeni, ki je trenutno noben proces ne obdeluje. Tako deluje BFS samo na nivoju domene. To implementacijo funkcije smo uporabili tudi pri pridobivanju strani za rezultate te seminarske naloge.

## 1.3 Spremembe podatkovne baze

V tabelo “data_type” smo dodali vrednosti: ZIP, RAR in ODT. V tabelo “page_type” smo dodali vrednosti: ERROR in NOTALOWED. ERROR predstavlja napako ob obisku strani. NOTALOWED predstavlja, da je bil obisk strani prepečen s strani robots.txt pravil.

# 2 Statistika

Pajka smo pognali z 10 procesi. V nekje 30 h delovanja je naš pajek obiskal 53.890 strani, v frontierju pa je bilo še 87.119 povezav. Število najdenih slik je 445.317, število binarnih datotek pa je 8.925. Število povezav med stranmi (vnosov v tabelo ’link’) pa je 1.438.989.

## 2.1 Zanimivosti

5 strani, ki vsebuje največ slik je:

Po pregledu zakaj te strani vsebujejo toliko slik smo ugotovili, da te strani vsebujejo veliko podvojenih slik npr. smeškoti v tabeli ali pa je slika uporabljena na več mestih kot gumb. Ker za slike ne preverjamo podvajanja, se vse te majhne sličice, ki so posejane po strani, zapišejo v tabelo ’image’, kar nato prinese tako velike številke.

| | |
|---|---|
| Število 30x preusmeritev | 2699 |
| Število 40x napak | 360 |
| Število 50x napak | 352 |
| Število strani označenih z ERROR | 715 |
| Število strani označenih z NOTALOWED | 2688 |

| | št. slik | URL |
|---|---|---|
| 1 | 921 | "http://kos.arso.gov.si/sl/indicators-trend" |
| 2 | 918 | "http://kazalci.arso.gov.si/sl/indicators-trend" |
| 3 | 766 | "http://meteo.arso.gov.si/uploads/probase/www/observ/surface/text/sl/observation_si/index.html" |
| 4 | 734 | "http://arsq.gov.si/Query/detail.aspx?ID=309270" |
| 5 | 700 | "https://www.fu.gov.si/podjetja/?cHash=af6dd4a718e107e8c44432bfd3c0201f&cp=1&tx_news_pi1%5B%40widget_0%5D%5BcurrentPage%5D=2" |

5 strani, ki vsebuje največ povezav je:

| | št. povezav | URL |
|---|---|---|
| 1 | 5814 | "https://podatki.gov.si/publisher" |
| 2 | 2001 | "http://eugo.gov.si/en/other-conditions/" |
| 3 | 1408 | "http://kos.arso.gov.si/sl/indicators-trend" |
| 4 | 1404 | "http://kazalci.arso.gov.si/sl/indicators-trend" |
| 5 | 1169 | "https://e-uprava.gov.si/drzava-in-druzba/javni-sektor/solstvo.html" |

Stran, ki najbolj izstopa po številu povezav je spletno kazalo, kjer so zapisane različne organizacije iz javnega sektorja.

# 3 Vizualizacija

Odločili smo se za vizualizacijo prikaza povezav med domenami (vnosi v tabelo 'site'). Vizualizacijo smo naredili s pomočjo knjižnjice D3.js. Izdelana vizualizacija je dinamična, saj se na statični vizualizaciji zelo težko vidi posamezne povezave. Dinamična izvedba je dosegljiva na povezavi: http://83.212.82.40/d3/