

Orodje za oddaljeno testiranje spletnih uporabniških vmesnikov

Jure Trilar¹, Vid Čermelj¹, Emilija Stojmenova Duh¹

¹Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška c. 25, Ljubljana
E-pošta: jure.trilar@fe.uni-lj.si

Remote online user interface testing tool

Abstract. *As the Covid-19 outbreak became an inevitable factor in our interface user-testing activities, we have devoted time and knowledge to implement an unmoderated remote online interface testing tool that corresponds with our requirements. In this article we describe inception, practical aspects of implementation, typical usage scenario and future prospects in development of such a tool. Special attention was given to producing mirrored copies of websites behind login forms to be included in interfaces to be tested with the tool.*

1 Uvod

Pandemija bolezni covid-19, ki se je v Sloveniji pojavila v drugem četrtletju 2020, nas je prisilila v razmislek, kako opraviti že načrtovane aktivnosti testiranja uporabniških vmesnikov, ki so do sedaj običajno potekala v laboratorijskem okolju skupaj z uporabniki. Ob ukrepih za preprečevanje okužb je prišlo spoznanje, da neposredni stiki z zunanjimi udeleženci testiranja v bližnji prihodnosti ne bodo mogoči zaradi spremenjenih družbenih navad in možnosti ponovitev valov okužb. Poleg tega pa neposredni stik z določenimi demografskimi skupinami, ki nas še posebej zanimajo, kot so npr. starejši, verjetno ne bo mogoč tudi v daljšem časovnem obdobju. Ne glede na to situacijo je potreba po izvedbi oddaljenih testov, ki omogočajo preprostejšo in udobnejšo udeležbo, evidentna zaradi po vsebini različnih, po strukturi pa podobnih in ponavljajočih se testiranj uporabniških vmesnikov [1].

Smiselna je investicija časa in znanja v to, da se vzpostavi infrastruktura za učinkovito oddaljeno testiranje uporabniških vmesnikov, ki bo obenem reševala še vrsto drugih težav, kot sta avtomatska obdelava anketnih podatkov brez ročnega vnašanja in natančno merjenje uporabniške izkušnje s standardnimi pristopi [2]. Pri tem obstaja tudi potencial za doseganje večjega vzorca testirancev. Vsekakor to prinaša določene metodološke zadržke, saj na primer ne bi mogli zaznati drugih, verbalnih in neverbalnih, signalov, prisotnih pri testiranju v laboratorijskem okolju, vendar so zadržki uravnoteženi s pomembnimi prednostmi takega merjenja uporabnosti, uporabniške izkušnje in obremenitev pri uporabi.

2 Izhodišča in izvedba orodja

V okviru raziskovalnega dela želimo testirati scenarije uporabe na posnetku portala eDavki [3] v dveh (originalnem in prilagojenem) uporabniških vmesnikih. Portal eDavki je javnosti dobro znana rešitev, ki ima pomembno vlogo v e-življenju in e-poslovnosti

slovenskih državljanov in za katero verjamemo, da bo podala relevantne rezultate. Primerjava rezultatov bo pokazala uporabnost koncepta družinsko usmerjenega razvoja (angl. family-centered design), ki ga razvijamo [4, 5].

Pri merjenju uporabnosti, uporabniške izkušnje in obremenitve pri opravljanju nalog se poslužujemo standardiziranih načinov merjenja (čas opravljanja nalog, uspešnost opravljanja nalog) in standardnih vprašalnikov (SUS [6], UEQ [7], NASA-TLX [8]), ki se uporabljajo na tem področju. Kombinacija vseh teh mehanizmov za merjenje ni bila združena v katerem od komercialnih orodij, ki so že na voljo [9]. Poleg tega nekatera obstoječa orodja sicer podpirajo večjezičnost, podpora za slovenski jezik pa ni navedena pri nobenem, zato smo se odločili za lastno izvedbo orodja.

Zaradi posebnih potreb glede zadostne možnosti prilagoditev in omenjenih tehničnih omejitev glede potreb testiranja [10] uporabniškega vmesnika portala eDavki smo identificirali nekaj izhodišč:

- Nemoderirano testiranje na daljavo. Udeleženci samostojno rešujejo naloge, brez osebe, ki vodi testiranje. Pri tem je ključno, da ne morejo zaiti izven »območja testiranja«, obenem pa se ne smejo znajti v brezizhodni situaciji, ki jim bi onemogočala nadaljevanje oz. zaključek testiranja, v nasprotnem primeru je treba to zaznati in ustrezno označiti v podatkih.

- Testiranje temelji na reševanju nalog in standardnih meritvah, ki se nanašajo na uspešnost in čas reševanja naloge. Udeleženci prejmejo navodila, nato rešujejo serijo nalog. Naloge konkretnega testiranja si v vsakem od vmesnikov sledijo linearno in obsegajo prijavo v zaledni del portala eDavki, iskanje vloge za obročno plačilo davka v največ treh mesečnih obrokih za fizične osebe in njeno oddajo. V snovanju scenarija smo predvidevali, da uporabniki niso že vnaprej seznanjeni s postopkom, kot je to npr. možno pri bolj popularni vlogi za popravek dohodnine. Izboljšave so na mestih, ki niso predmet redne uporabe, temveč jih uporabniki obiščejo redko, npr. enkrat letno ali manj. S tem ni učinka učenja in je večja potreba po še bolj intuitivni uporabi. Zavedamo se, da izbrani scenariji le delno naslovijo testiranje koncepta družinsko usmerjenega razvoja.

- Interaktivnost in prilagodljivost vmesnikov. Testirani posnetki vmesnikov morajo delovati kot v originalu. Omogočati morajo polno uporabniško izkušnjo, obenem pa mora obstajati možnost prilagoditve delovanja za testiranje posebnih mehanizmov prikaza in interakcije. V prvi iteraciji je bila rešitev razvita za namizne računalnike, z vmesniki zaslona, miške in tipkovnice, v prihodnosti pa bi bila pomembna izboljšava testiranja z mobilnimi napravami.

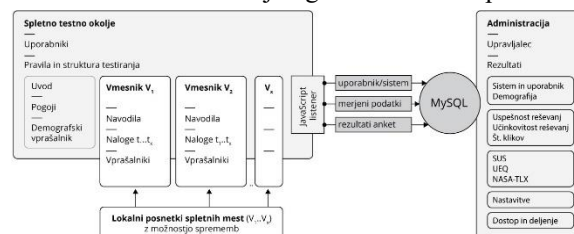
- Upoštevanje dobrih praks in standardnih postopkov testiranja uporabniških vmesnikov. Primer: izničiti je treba učinek učljivosti [2], zato se testirani vmesniki in naloge, kjer je to potrebno, pojavijo v naključnem vrstnem redu. Za testiranje uporabljamo standardizirane metrike s področja raziskovanja uporabniške izkušnje.
 - Zasebnost in upoštevanje aktualnih pravil uporabe osebnih podatkov sta obvezna.
 - Po-uporabljivost orodja za druge raziskovalce in druge projekte.
- Navedena izhodišča so služila za prvotno načrtovanje okolja za oddaljeno testiranje in pomenijo prvotne okvire za izvedbo. To niso kritične omejitve in orodje bo v prihodnosti prilagojeno za morebitne nove potrebe.

2.1 Izvedba

Prototip orodja je izdelan s spletnimi jeziki HTML5, CSS3, Javascript in PHP z bazami MySQL, to je kombinacija, ki ustreza splošnim oblikam komercialnih gostovanj v Sloveniji. Orodje je sestavljeno iz treh funkcionalnih komponent (Slika 1):

- Uporabniški del je okolje, namenjeno testnim uporabnikom, ki jih povabimo k reševanju z enotnim URL-naslovom. V tem okolju uporabniki dobijo navodila za reševanje nalog, ki se prikazujejo na različnih testnih vmesnikih v naključnem vrstnem redu. Testni vmesniki, posnetki obstoječih spletnih strani ali namenski prototipi so »vstavljeni« v poseben element na spletni strani. Ob upoštevanju varne povezave (https) in povezanih priporočil [11, 12] glede uporabe zunanjih virov posameznih brskalnikov dobimo možnosti manipuliranja HTML-strukture z vsiljeno kodo Javascript. To je ključno za spremljanje ciljnih akcij, ki so praviloma v obliki klika na v nastavitvah določen element vmesnika. O uporabnikih zberemo splošne demografske podatke in poskušamo zagotoviti istovetnost posameznega reševalca skozi t. i. *device/browser fingerprint*, brez uporabe piškotkov brskalnika. To storimo za primer poskusov zlorabe in neprimerne vplivanja na rezultate.
- Posnetki vmesnikov in prototipni vmesniki so pomembni za kvalitetno izvedbo testiranja, zato temu namenimo samostojno poglavje. V testno okolje so integrirani kot samostojne enote, ki omogočajo, da uporabnik vidi ter uporablja vmesnike na način, identičen originalom, kadar gre za posnetek obstoječe spletne strani oz. aplikacije, ali pa dejanski prototip brez večjih omejitev testiramo v istem okolju. Možno je testiranje več vmesnikov. Treba je opozoriti, da je kvaliteta posnetka odvisna od zunanjega orodja, ki ga uporabljamo za ta namen. Različna orodja namreč ponujajo različne pristope za zajem spletnih strani, portalov ali spletnih aplikacij.
- Administrativni del je na voljo le lastniku infrastrukture. Prikazuje vse zbrane podatke o reševanju nalog, izračunane rezultate vprašalnikov, omogoča nastavitve opravljanja nalog, shranjevanja podatkov in parametrov posameznih testnih vmesnikov. V začetni iteraciji še ni na voljo mehanizem za upravljanje z

uporabniki. Prav tako še niso na voljo vse načrtovane funkcionalnosti. V prihodnosti bo tako na voljo tudi priporočilno orodje, ki zbrane podatke posameznih vmesnikov primerja glede na določene pristope k zasnovi uporabniških vmesnikov. Dodatne funkcije in izpisi rezultatov se bodo dodajali glede na bodoče potrebe.



Slika 1. Shema zasnove

2.2 Namestitve

Testno okolje je sestavljeno iz najbolj pogoste konfiguracije slovenskih komercialnih ponudnikov gostovanj – t. j. Apache, PHP, MySQL na strežniški strani. Testno okolje je moč vzpostaviti tudi na lastni infrastrukturi.

Testne vmesnike v obliki posnetkov obstoječih strani ali prototipov prenesemo v relevantno podmapo v direktorjski strukturi. Namestimo lahko enega ali več testnih vmesnikov – če jih je več, uporabnik testira serijo vmesnikov v naključnem vrstnem redu zaradi izogibanja učinku priučljivosti.

Posebno pozornost je treba nameniti nastavitvam, ki jih trenutno shranjujemo v naboru parametrov. Nastaviti je treba pot do posameznega vmesnika v direktorjski strukturi. Določimo serije ciljnih akcij – praviloma gre za klike na določene elemente v testiranih vmesnikih. V testnih vmesnikih moramo ročno poimenovati takšne elemente z oznakami razredov (angl. class) ali identifikatorji (angl. ID). Pri tem določimo tudi opis s ciljno akcijo povezane naloge in polje shranjevanja v bazi MySQL. Nastavimo tudi čas, po izteku katerega se uporabniku omogoči preskočiti nalogo, in pot nadaljevanja testiranja v naslednjem koraku. Čas nastavimo glede na lastno oceno kompleksnosti vmesnika in zgornje meje, za katero menimo, da začne pri uporabniku povzročati velike težave z upravljanjem vmesnika. Vsekakor ima uporabnik še vedno možnost zaključiti ciljno akcijo naloge; če se izgubi ali meni, da je naloga pretežka, pa jo preskoči in v bazo se doda zapis o neuspešnosti opravljanja naloge.

3 Posnetki obstoječih uporabniških vmesnikov

Ključna komponenta za izvedbo testiranja obstoječih uporabniških vmesnikov so t. i. posnetki spletnih strani (angl. website mirrors), ki omogočajo enako uporabniško izkušnjo kot original, vendar delujejo popolnoma samostojno. Omogočajo testiranje kopij vmesnikov, novih funkcij na teh vmesnikih v testnem okolju in testiranje samostojnih prototipnih rešitev – ter neposredno primerjavo med različnimi vmesniki, kot na

primer pri t. i. testiranju A/B [13]. Za pridobitev posnetkov obstoječih spletnih strani je treba previdno izbrati pravi način izvedbe, saj hitro naletimo na vrsto tehničnih in drugih izzivov. Za avtomatiziran zajem spletnih vmesnikov praviloma uporabljamo t. i. pajke, ki jih lahko naredimo sami, lahko pa se poslužimo že izdelanih rešitev.

Pri izbiri orodja oziroma pisanju programa za pridobivanje posnetkov spletnih strani je treba posebno pozornost nameniti robustnosti in vljudnosti spletnega pajka.

Robustnost: Med prehajanjem po spletnih straneh se lahko pajek ulovi v past (angl. spider trap). S tem se lahko zaplete v neskončno indeksiranje istih strani znotraj neke domene. Vse pasti niso nujno zlonamerne, do zapletanja lahko pride tudi v primeru, ko spletna stran ni bila pravilno oblikovana.

Vljudnost: Spletni strežniki imajo različne politike glede spletnih pajkov. V njih so zapisana različna priporočila, ki jih je treba pri obisku in zajemu teh strani upoštevati. Tudi če spletne strani nimajo definirane politike za spletne pajke, je priporočljivo, da se držimo dobrih praks [14].

3.1 Dobre prakse uporabe spletnih pajkov

Pri uporabi spletnih pajkov obstajajo dobre prakse, ki se jih držijo vsi največji iskalniki in so implementirane tudi v večini programov za ekstrakcijo podatkov s spleta. Po potrebi jih je mogoče vklopiti ali izklopiti. Razdelimo jih lahko v tri sklope pravil:

Spletni pajek se mora predstaviti kot spletni pajek in se ne sme pretvarjati, da je navaden uporabnik.

Spletne strani beležijo število obiskovalcev, poleg tega pa jim to omogoči nadzor nad pasovno širino, ki jo namenijo spletnih pajkom.

Spletni pajek mora spoštovati pravila datoteke robots.txt, ki določa, do katerih strani ima pajek dostop.

Upravljalci s tem sporočijo, na katerih straneh nočejo imeti spletnih pajkov, in tudi posredno potrdijo, da so spletni pajki na preostalih straneh v isti domeni dovoljeni. Ta datoteka pajku ne prepreči dostopa in jo lahko ignorira, vendar je pri dostopanju do takih spletnih strani treba biti pozoren, saj lahko vsebujejo podatke, ki jih je prepovedano hraniti (osebni podatki, avtorska dela). Če nas podatki na teh straneh ne zanimajo, je bolje, da jih ne shranjujemo.

Spletni pajek na posamezni spletni strani ne sme porabiti preveč pasovne širine.

To pomeni, da s posamezne strani ne prenaša več kot ene datoteke hkrati in da med posameznimi prenosi počaka nekaj sekund. Če shranjujemo podatke z različnih domen, je bolje, da vsakič obiščemo drugo spletno stran, ki jo izberemo z algoritmom krožnega izbiranja (angl. round-robin) [15, 16].

3.2 Izbira in uporaba orodja za posnetke uporabniških vmesnikov

Za potrebe izdelave posnetkov obstoječih spletnih strani smo iskali primerno obstoječe orodje [17]. Med vsemi, komercialnimi in odprtokodnimi, ki smo jih preizkusili, se je izkazalo, da našim potrebam najbolj ustreza odprtokodno orodje Htrack [18], ker ponuja najširši

spekter možnih nastavitvev pri tem opravilu, izdeluje uporabne posnetke, ki dobro delujejo v nepovezanem (*offline*) načinu, ter, najpomembnejše, omogoča snemanje strani, ki so dostopne le ob prijavi uporabnikov v zasebne dele strani, kar je ključno pri pridobivanju kopij spletnega portala eDavki.

Nabor možnosti pri upravljanju te rešitve je izredno dolg in podrobno opisan na straneh z dokumentacijo [19]. Posebej pa je treba omeniti možnost, da po prijavi z uporabniškim imenom in geslom v zasebne dele spletnih portalov izvozimo ustrezne piškotke. Te, shranjene v formatu Netscape, namestimo v ciljno mapo in orodje Htrack samodejno upošteva prijave, ki so dostopne prek piškotkov, ter naredi kopije zasebnih delov strani. Od upravljalca portala eDavki smo dobili dovoljenje za pridobitev posnetkov zasebnega dela uporabniškega vmesnika za potrebe raziskave, v okviru katere bomo v izvajali testiranje uporabnosti koncepta družinsko usmerjenega razvoja.

4 Uporaba orodja in prikaz rezultatov

4.1 Tipični scenarij uporabe

Po namestitvi okolja in nastavitvi različnih parametrov vmesnika z enotnim URL-naslovom povabimo udeležence k sodelovanju. Vabilo diseminiramo v skladu s cilji raziskave in želenim ciljnim vzorcem.

Uporabnikom, ki sledijo povezavi, se v začetnih navodilih predstavijo cilji in namen raziskave, odgovorna oseba in kateri podatki se zbirajo. Po GDPR je shranitev podatkov o internetnem naslovu (angl. IP address) ter podpisu naprave in brskalnika (t. i. tehnika *browser/device fingerprinting*) dovoljena [20] le ob izrecnem dovoljenju testiranega uporabnika; čeprav se na prvi pogled zdi, da takšni podatki ne vsebujejo osebnih podatkov.

Testiranje oz. beleženje se začne po dovoljenju za zbiranje podatkov. Ob tem se označi čas začetka testiranja. Uporabnik nato reši preprost demografski vprašalnik, ki se nanaša na cilje raziskave.

Sledijo navodila za reševanje serije nalog. Opis trenutne naloge je prikazan v vmesniku. Naloga velja za uspešno rešeno, če uporabnik izvede ciljno akcijo – ki je lahko kjerkoli v strukturi vmesnika. Če od začetka reševanja naloge preteče arbitrarno določen čas, se uporabniku ponudi možnost, da trenutno nalogo preskoči, sistem pa to ustrezno zabeleži – na ta način poskušamo povečati možnost, da uporabnik nadaljuje s testiranjem, in s tem zagotovimo večji nabor podatkov.

Na koncu vsakega sklopa nalog v vmesniku sekvenčno sledijo standardni vprašalniki o uporabniški izkušnji, uporabnosti sistema in obremenitvi pri upravljanju nalog. Sledi še zahvala testnim uporabnikom za njihov prispevek k raziskavi in zabeležba v bazi o končanju testiranja.

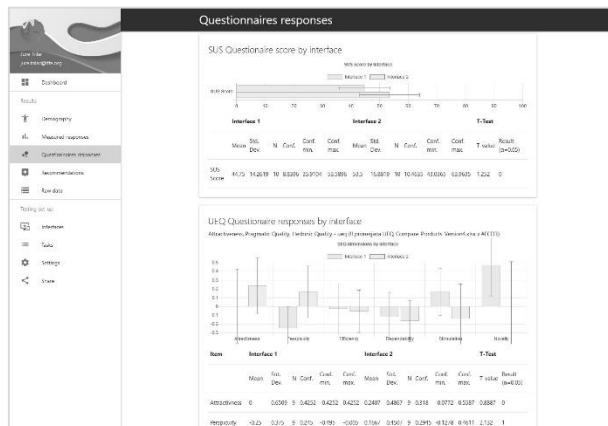
4.2 Izpis rezultatov

Kumulativne rezultate testiranja izpišemo v administratorskem vmesniku in obsegajo primerjave merjenih vrednosti med različnimi testnimi vmesniki na nivoju nalog, kjer prikazemo povprečno uspešnost in

popravljen čas reševanja posamezne naloge ter dodatno še povprečno število klikov ob tem.

Prikažemo tudi izračunane vrednosti vprašalnikov, ki temeljijo na subjektivni oceni uporabniške izkušnje, uporabnosti in napora pri reševanju nalog (Slika 2). Prikažemo jih na način, ki je lasten vsaki metodi, npr. porazdelitev po komponentah uporabniške izkušnje ali pa rezultat na enotni lestvici uporabnosti.

Izpis rezultatov je prilagojen trenutnim potrebam raziskave vmesnikov, ki temeljijo na portalu eDavki.



Slika 2. Zaledni vmesnik in prikaz rezultatov

5 Zaključek

Pri našem dosedanem delu je testiranje vmesnikov potekalo v laboratorijskem okolju. Zato je treba obravnavati določene zadrege, ki nastanejo v moderiranem in (za nas novem) nemoderiranem načinu v zvezi z vplivi na rezultat. Obstaja možnost, da udeleženci prilagodijo svoje odgovore (npr. prava starost, digitalne veščine), če je prisotna oseba, ki vodi testiranje. V nemoderiranem načinu ne moremo učinkovito zajeti nekaterih drugih signalov, besedne in nebesedne komunikacije udeležencev, ki nam daje globlji, kvalitativni vpogled v uporabnost, uporabniško izkušnjo in obremenitve pri uporabi sistemov ter nam kadarkoli omogoča testiranje na večjem vzorcu. V to je treba vključiti razmisleke o kombiniranem pristopu, kjer s sledenjem kurzorju miške ali z uporabo video spletne konference med reševanjem nalog poskusimo zaznati dodatne značilnosti uporabe testiranih vmesnikov. Primerjava rezultatov med moderiranim in nemoderiranim pristopom s pomočjo manjše skupine testirancev bi lahko pokazala ustreznost ali pomanjkljivosti orodja za oddaljeno testiranje. Dodatna izboljšava bi bil razmisleki glede tega, kako vizualno motivirati uporabnike za reševanje.

Razumljivo je, da testno okolje, ki je bilo izvedeno, ni konkurenca obstoječim komercialnim rešitvam – vsekakor pa dobro služi našim, bolj akademskim, potrebam pri oddaljenem testiranju uporabniških vmesnikov in ga bomo prilagajali oz. dopolnjevali glede na cilje prihodnjih raziskav.

6 Literatura

- [1] ISO 9241-11:2018 – Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts.
- [2] T. Tullis, B. Albert, Measuring the User Experience: Collecting, Analyzing and Presenting Usability Metrics, Morgan Kaufmann, Amsterdam, 2016.
- [3] eDavki. Dosegljivo: <https://edavki.durs.si/> [6. 7. 2020].
- [4] J. Trilar, A. Kos, S. Jazbinšek, L. Jensterle, in E. S. Duh, ICT to Promote Well-Being within Families, *Sensors*, vol. 18, no. 9, p. 2760, 2018.
- [5] J. Trilar, V. Zavratnik, V. Čermelj, B. Hrast, A. Kos, in E. S. Duh, Rethinking Family-Centred Design Approach Towards Creating Digital Products and Services, *Sensors*, vol. 19, no. 5, p. 1232, 2019.
- [6] K. Finstad, The Usability Metric for User Experience, *Interacting with Computers*, vol. 22, no. 5, 2010, str. 323–327, doi:10.1016/j.intcom.2010.04.004.
- [7] B. Laugwitz, T. Held, and M. Schrepp, Construction and Evaluation of a User Experience Questionnaire, *Lecture Notes in Computer Science HCI and Usability for Education and Work*, pp. 63–76, 2008.
- [8] A. Cao, K. Chintamani, A. K. Pandya in R. D. Ellis, NASA TLX: Software for assessing subjective mental workload, *Behavior Research Methods*, vol. 41, no. 1, pp. 113–117, 2009.
- [9] Justinmind, Best 15 usability testing tools for happy users. Dosegljivo: <https://www.justinmind.com/blog/usability-testing-tools/> [6. 7. 2020].
- [10] Nielsen Norman Group, Tools for Unmoderated Usability Testing. Dosegljivo: <https://www.nngroup.com/articles/unmoderated-user-testing-tools/> [6. 7. 2020].
- [11] MDN Web Docs, Same-origin policy. Dosegljivo: https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy [6. 7. 2020].
- [12] Wikipedia, Cross-origin resource sharing. Dosegljivo: https://en.wikipedia.org/wiki/Cross-origin_resource_sharing [6. 7. 2020].
- [13] Optimizely, A/B testing. Dosegljivo: <https://www.optimizely.com/optimization-glossary/ab-testing/> [6. 7. 2020].
- [14] C. D. Manning, P. Raghavan, H. Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008, pp. 405–420.
- [15] B. Ribeiro, R. Baeza-Yates, *Modern Information Retrieval*, Upper Saddle River, NJ: Pearson Higher Education, 2010, pp. 515–544.
- [16] R. Mitchell, *Web Scraping with Python: Collecting More Data from the Modern Web*, O'Reilly Media, Incorporated, 2018, pp. 263–278.
- [17] Top 15 Website Ripper or Website Downloader Compared. Dosegljivo: <https://prowebscraper.com/blog/top-website-ripper-or-website-downloader-compared/> [6. 7. 2020].
- [18] HTTrack Website Copier, Dosegljivo: <http://www.httrack.com/> [6. 7. 2020].
- [19] HTTrack Help, Dosegljivo: <http://httrack.kauler.com/help/Home> [6. 7. 2020].
- [20] K. Szymielewicz, B. Budington, The GDPR and Browser Fingerprinting: How It Changes the Game for the Sneakiest Web Trackers, Electronic Frontier Foundation. Dosegljivo: <https://www.eff.org/deeplinks/2018/06/gdpr-and-browser-fingerprinting-how-it-changes-game-sneakiest-web-trackers> [6. 7. 2020].