Testiranje stranice Links

Uvod

Projekt se temelji na automatskom testiranju stranice Links, a to se ostvaruje korištenjem Node.js-a te biblioteke puppeteer. Node.js predstavlja Javascript okruženje za izvršavanje Javscript koda izvan preglednika. Za testiranje web stranice unutar Node.js projekta instalirana je biblioteka puppeteer. Ona za razliku od Seleniuma, koji je trenutno najrašireniji alat za automatsko testiranje, podržava samo Google Chrome. No, kada je u pitanju brzina izvršavanja samih testova, puppeteer daje značajno bolje rezultate od Seleniuma.

Opis automatskog testa

Za početak, kako bi se Chrome pokrenuo u vidljivom načinu rada potrebno je opciji headless pridružiti vrijednost false, a to je prikazano na slici 1. Uz headless opciju predane su još dvije opcije: slowMo i args. Opcija slowMo omogućuje usporavanje radnji puppeteera za predani broj milisekundi, a pomoću args se spriječava iskakanje notifikacija.

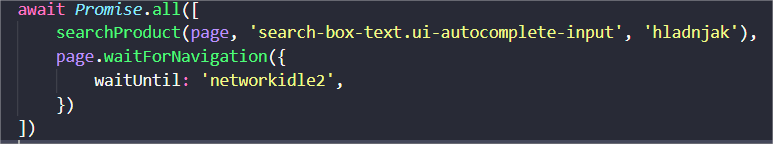


Slika 1. Pokretanje Chroma u vidljivom načinu rada

Nakon pokretanja Chroma otvara se novi tab i podešava se širina i visina zaslona. One su prilagođene za zaslon veličine 15.6″ te ako je vaš zaslon drugačiji od navedenog, preporučava se odlazak na ovu stranicu: http://whatismyscreenresolution.net/ te izmjena druge linije koda koja je prikazana na slici 2.



Slika 2. Otvaranje novog taba i podešavanje zaslona



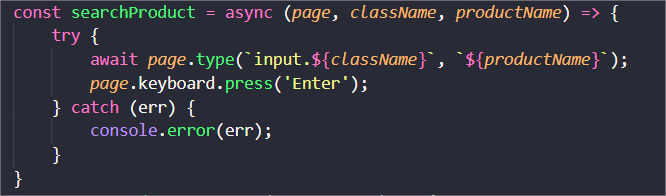
Slika 3. Poziv funkcije za unos teksta

Slika 3 prikazuje pozivanje funkcije koja je zadužena za unos teksta u tražilicu. Funkciji se osim paga predaje ime klase i proizvoda. Ime klase je jedinstvena oznaka elementa, a do te oznake se dolazi tako da se "inspecta" stranica, vidi sliku 4.



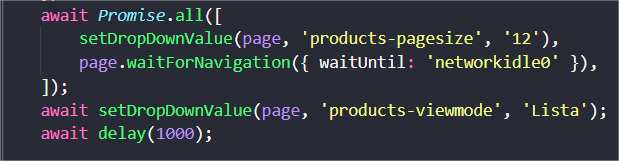
Slika 4. “Inspect” na tražilici

Na slici 5 prikazano je upisivanje proizvoda u tražilicu te pritisak tipke enter.



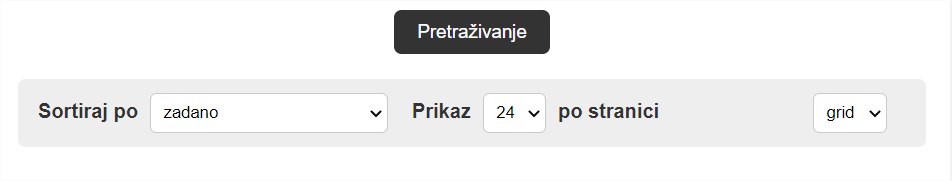
Slika 5. Pretraga proizvoda

Nakon pretrage proizvoda čeka se potpuno učitavanje stranice te se zatim testiraju padajući izbornici, a pozivi funkcija prikazani su na slici 6.



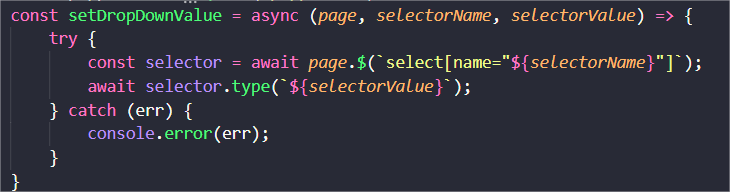
Slika 6. Poziv funkcija za testiranje padajućih izbornika

Testiraju se dva padajuća izbornik, a to su prikaz po stranici i način prikaza, vidi sliku 7.



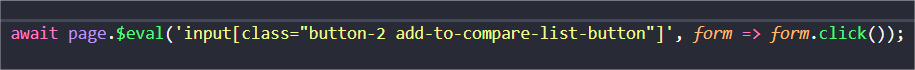
Slika 7. Padajući izbornici

Funkcija koja testira padajuće izbornike, prvo pronalazi izbornik po imenu te zatim u njega upisuje određenu vrijednost, a to vidimo na slici 8.



Slika 8. Funkcija za testiranje padajućih izbornika

Nakon testiranja padajućih izbornika, prva dva proizvoda dodaju se u listu za usporedbu, a to se postiže pozivom linije sa slike 9 dva puta.



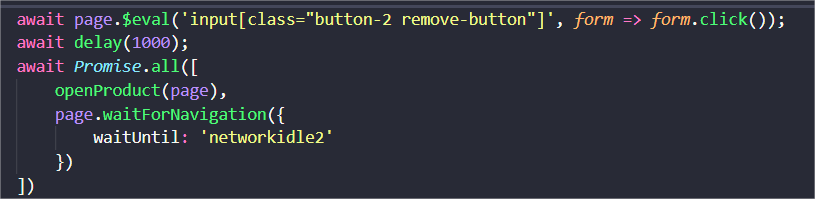
Slika 9. Usporedba proizvoda.

Nakon što su dodani proizvodi, pritišće se gumb „Usporedi proizvode“, vidi sliku 10.



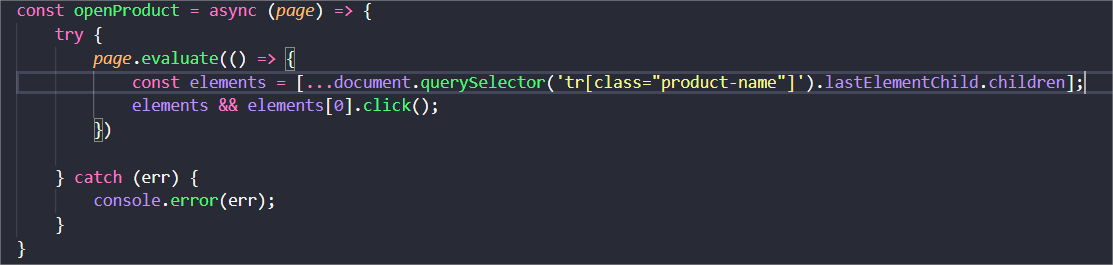
Slika 10. Odlazak u usporedbu proizvoda

Na slici 11. prikazano je brisanje prvog proizvoda iz liste za usporedbu te zatim otvaranje drugog proizvoda.



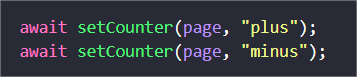
Slika 11. Brisanje prvog te otvaranje drugog prozivoda

Otvaranje proizvoda se izvršava na način da se pritisne ime koje je ništa drugo nego link. No, link nema pridodjeljenu klasu te je zbog toga potrebno koristit querySelector za njegov pronalazak.

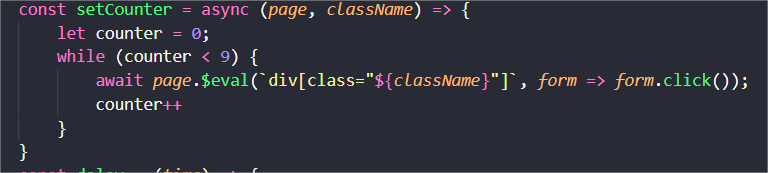


Slika 12. Pronalazak linka

Nakon što se proizvod otvori, testiraju se gumbovi za povećavanje i smanjivanje količine prozivoda. Prvo se poziva gumb koji će povećati količinu na 10 te se zatim poziva gumb koji će ju vratiti na početnu vrijednost.



Slika 13. Poziv funkcija za postavljanje količine proizvoda



Slika 14. Logika za postavljanje količine proizvoda

Na kraju testa proizvod se dodaje u košaricu te se nakon dodavanja odlazi u nju. Preglednik ostaje upaljen te se unutar terminala ispisuje poruka Test passed.

Slika 15. Dodavanje proizvoda u košaricu te odlazak u nju