Pannon Egyetem

Műszaki Informatikai Kar

Rendszer- és Számítástudományi Tanszék

programtervező informatikus BSc

SZAKDOLGOZAT

Kotlin web scraping fejlesztése

Radó Bence

Témavezető: Machalik Károly

2020

Alulírott Radó Bence hallgató, kijelentem, hogy a dolgozatot a Pannon Egyetem Rendszer- és Számítástudományi Tanszék tanszékén készítettem a programtervező informatikus végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozatban lévő érdemi rész saját munkám eredménye, az érdemi részen kívül csak a hivatkozott forrásokat (szakirodalom, eszközök, stb.) használtam fel.

Tudomásul veszem, hogy a dolgozatban foglalt eredményeket a Pannon Egyetem, valamint a feladatot kiíró szervezeti egység saját céljaira szabadon felhasználhatja.

Veszprém, 2020. november 1.

aláírás

Alulírott Machalik Károly témavezető kijelentem, hogy a dolgozatot Radó Bence a Pannon Egyetem Rendszer- és Számítástudományi Tanszék tanszékén készítette programtervező informatikus végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozat védésre bocsátását engedélyezem.

Veszprém, 2020 november 1.

aláírás

Köszönetet szeretnék nyilvánítani Machalik Károlynak, aki végig segítette munkámat, illetve szakmai tanácsokkal látott el, a Pannon Egyetem Tanárainak akiktől a megfelelő tudást elsajátíthattam az évek során, továbbá Édesanyámnak és Édesapámnak akik támogattak tanulmányaim során és bíztak bennem.

**TARTALMI ÖSSZEFOGLALÓ**

A dolgozatom témája amit választottam: Kotlin web scraping fejlesztése. A web scraping egy gyors és könnyű módja különböző adatok kinyerésének különböző weboldalakról. Nagyon sokféle web scraping program létezik, viszont mindegyiknek meg van az előnye, illetve a hátrányais. Témám során a Kotlin nyelvet használtam, és arra törekedtem, hogy a legkönnyebben használható és testreszabható programot készítsek el ami mindenki számára könnyen használható. [TODO]

**Kulcsszavak:** Web, Kotlin, Scraping, Adat

**TARTALOMJEGYZÉK**

[TODO]

**TARTALMI RÉSZ**

**Bevezetés**

A feladat címe amit választottam a következő: Kotlin web scraping fejlesztése. Ez annyit jelent, hogy Kotlin nyelven kell megvalósítani az adott programot, ami jelen esetben egy web scraping alkalmazás. Először a Kotlin nyelvről szeretnék picit írni. A Kotlin erősen típusos, objektumorientált programozási nyelv. Szentpétervár közelében lévő Kotlin-szigetről nevezték el. 2011-ben hozták nyilvánosságra a nyelv létezését. A JAVA nyelvvel szokták összehasonlítani. A JAVA-val teljes mértékben kompatibilis a Kotlin, továbbá könnyebben tanulható, mivel egyszerűbb a szintaxisa. A web scraping-et tulajdonképpen arra használják, hogy különböző weboldalakról különböző adatokat nyernek/gyűjtenek ki, hogy ezeket rendszerezetten egy helyen tárolják, majd később fel tudják használni különböző tevékenységekhez. Nevezik még web harvesting illetve web data extraction-nek is. Ezt a folyamatot úgy kell elképzelni, mintha egy személy felmenne egy weboldalra és kézzel kimásolgatna különböző adatokat a weboldalról és kigyűjtené magának. A web scraping-gel, rengeteg időt és sok fáradalmat lehet megspórolni az automatizáció miatt. Léteznek olyan web scraping programok amelyekhez szükséges külső interakció, vagyis a felhasználónak különböző paramétereket kell megadnia a programnak, például, hogy milyen típusú adatokat keressen a program az adott weboldalon. Azonban léteznek olyanok is, amik teljesen automatizáltan gyűjtik ki az adatokat és a háttérben futnak. Egy picit arról is írnék, hogy miért is választottam ezt a témát. A mobil programozás mindig is foglalkoztatott, ennek kapcsán elkezdett érdekelni a JAVA nyelv, mivel az Android alkalmazások nagyrésze JAVA-ban van írva. Azonban pár éve bejelentette a Google, hogy az elsődlegesen támogatott nyelv a Kotlin lesz. Ez volt a fő oka annak, hogy ezt a témát választottam. A másik pedig, hogy ki szerettem volna próbálni magam egy tőlem elég távol álló területen, mint például a webbel kapcsolatos problémák megoldása. Nagyon sok web scraping program található az interneten. Léteznek ingyenesen használhatóak, vannak amelyekért fizetni kell, teljesen automatizáltak illetve olyanok is amelyekhez felhasználói beavatkozás szükséges. A szakdolgozatom végére azt szeretném elérni, hogy egy olyan programot készítsek, ami ingyenesen elérhető bárki számára és könnyen használható. Egy nagyon letisztult GUI-t képzelek el a program számára, hogy minél könnyebben bele tudjanak jönni a felhasználók a használatába. A legnagyobb probléma amivel találkoztam a különböző web scraping programok tanulmányozása során, az az volt, hogy nem voltak testreszabhatóak, ezt úgy értem, hogy nagyrészük egy adott funkciót tudott. Illetve, hogy egy adott ID-jű mezőnek az ID-jét gyűjtötték ki, azonban egy dinamikusan változó rendszerben ez mindig változik. Az én programomban úgy próbálom kiküszöbölni ezeket a problémákat, hogy testreszabható legyen a program. Ezt úgy értem, hogy meg lehet majd adni, hogy milyen típusú, ID-jű mezőket, illetve milyen reguláris kifejezésekre keressen rá a program és azok alapján gyűjtsön adatokat. Olyan funkciót is szeretnék a programba, aminek segítségével lehet exportálni a kigyűjtött adatokat egy Excel fájlba, hogy azokat rendszerezve egy helyen lássa a felhasználó, hogy majd később könnyebben fel lehessen használni őket.

**Kutatás**