Univerzita Pavla Jozefa Šafárika v Košiciach

Prírodovedecká fakulta

**WEBOVÝ SCRAPER V ROZŠÍRENÍ PREHLIADAČA S POLOAUTOMATICKOU ANOTÁCIOU**

Analýza a návrh riešenia

Bc. Dávid Varga

**Študijný program:** Informatika

**Školiace pracovisko:** Ústav informatiky

**Vedúci záverečnej práce:** RNDr. Peter Gurský, PhD.

**Košice 2019**

# Hlavné prístupy anotovania

V tejto kapitole si rozdelíme scraper-e do troch hlavných skupín podľa prístupu anotovania, ktoré sme identifikovali počas testovania týchto scraper-ov. Niektoré scraper-e sa nachádzajú vo viacerých skupinách, pretože tieto prístupy kombinujú.

## Anotovanie pomocou uložených schém

Schémy sú niečo ako vzory, podľa ktorých vieme anotovať rôzne typy objektov, či už viacero zobrazených na jednej stránke, alebo objektov, ktorých informácie sa nachádzajú na viacerých stránkach. Tieto schémy je možné editovať alebo vytvárať nové, čím je možné vytvoriť si schému podľa vlastných potrieb. Ak nastane situácia, že bude potrebné anotovať viacero objektov z nejakého zoznamu a pre každý objekt budeme chcieť prejsť na jeho detailovú stránku, tak je možné dať do schémy typ atribútu „klik“. Tento typ atribútu prepojí schému pre detailovú stránku objektu s pôvodnou schémou.

### Zoznam scraper-ov využívajúcich uložené schémy

Connotate, Agenty, Content Grabber, Data Miner, Data Toolbar, Easy Web Extract, Helium Scraper.

## Anotovanie pomocou postupného vytvárania pravidiel

Pri tomto spôsobe anotovania si scraper postupne spolu s používateľom vytvára sadu pravidiel pre anotovanie objektov. Zároveň sa vytvárajú schémy, avšak nie sú tak univerzálne, aby sa dali použiť na rôznych stránkach súčasne. Taktiež sa takto vytvorené pravidlá väčšinou uložia ako projekt prislúchajúci k webovej stránke. Niektoré scraper-y využívajúce tento spôsob sú veľmi používateľsky prívetivé, avšak obsahujú menej funkcionalít. Pre používateľov, ktorí nepotrebovali často anotovať zložité objekty, bol tento spôsob najvhodnejší. Programy Data Miner a Mozenda popri anotovaní veľmi kvalitne popisujú, čo má používateľ robiť a vedú ho tak, že nie je potrebné študovať tutoriály alebo návody k týmto scraper-om.

### Zoznam scraper-ov využívajúcich postupné vytváranie pravidiel

ParseHub, Agenty, Content Grabber, Data Miner, Data Toolbar, Dexi.io, Easy Web Extract, FMiner, GetData.IO, Grepsr, Helium Scraper, Import.IO, Mozenda, Visual Web Ripper, Web Scraper, Web Sundew, Scrape.it, Web Content Extractor, WebHarvy.

## Anotovanie pomocou umelej inteligencie

Tento spôsob anotovania využíva umelú inteligenciu. Prvý krok je vloženie URL adresy webovej stránky, ktorá má byť extrahovaná. Umelá inteligencia automaticky anotuje objekty vyskutujúce sa na danej webovej stránke a vloží ich do tabuľky. V prípade, ak sú objekty zle anotované, tak scraper Import.io môže využiť svoju funkcionalitu úpravy vytvorených pravidiel pre extrakciu. Po automatickej anotácii si vie používateľ tieto pravidlá vyladiť podľa vlastných potrieb. Pravidlá vytvorené scraper-om Instant Data Scraper nie je možné upravovať a používateľ k nim nemá prístup. Používateľ môže zvoliť možnosť vygenerovania iných pravidiel, čím získa inú sadu pravidiel pre anotovanie, ktorá môže presnejšie anotovať objekty. Tento spôsob často vedie k nepostačujúcim výsledkom.

### Zoznam scraper-ov využívajúcich umelú inteligenciu

Import.io, Instant Data Scraper.

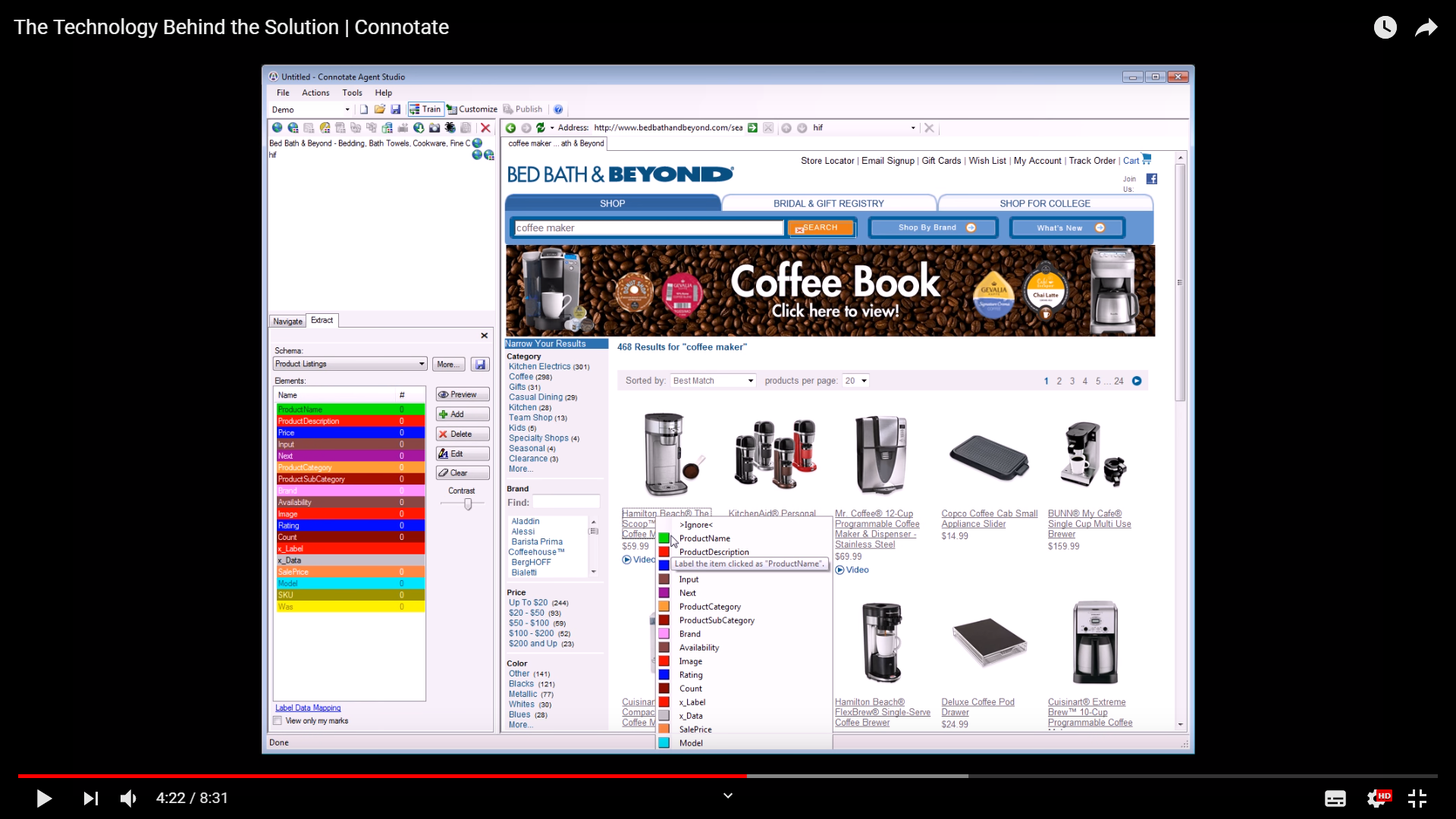
# Anotovanie zoznamu objektov

V praxi sa veľmi často vyskytuje, že objekty spolu s ich atribútmi sa nachádzajú v nejakom zozname. Ako zoznam objektov si môžeme predstaviť napríklad zoznam produktov v nejakom e-shop-e, pričom pri každom objekte je jeho názov, cena a krátky popis.

## Connotate

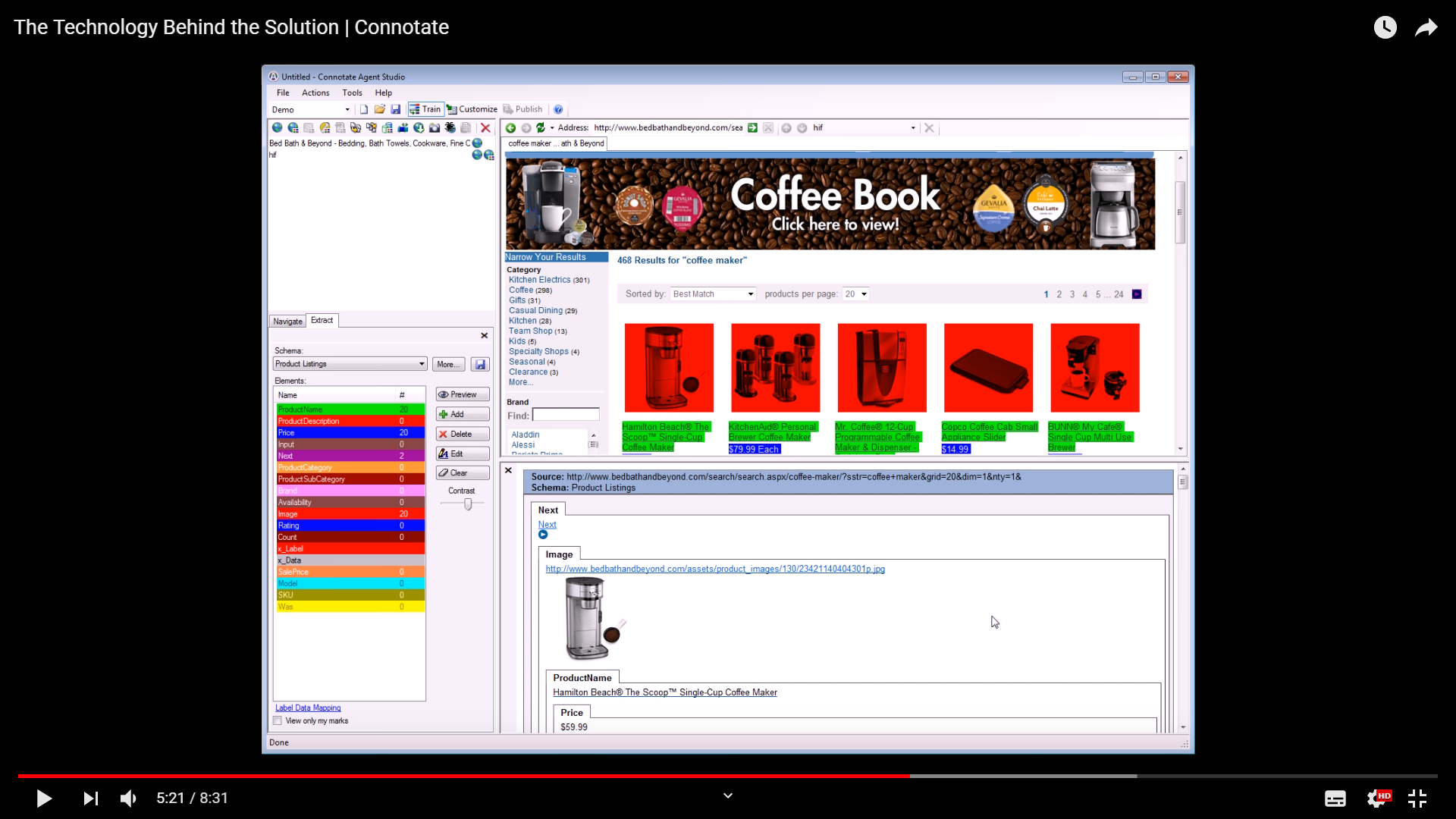
Pred začiatkom akéhokoľvek anotovania v programe Connotate si používateľ vyberie zo zoznamu hotových schém. Ak chce anotovať viacero objektov na jednej stránke, tak si vyberie schému, ktorá túto možnosť podporuje.

Na nasledujúcom obrázku je zoznam kávovarov na internetovom obchode [www.bedbathandbeyond.com](http://www.bedbathandbeyond.com). Používateľ sa snaží anotovať tieto kávovary, pričom chce získať cenu, nadpis a obrázok kávovaru (objektu). Vybral si schému Product Listings, ktorá vyhovuje pre anotovanie kávovarov alebo rôznych produktov nachádzajúcich sa na jednej stránke. Používateľ následne potrebuje určiť, kde sa nachádzajú vlastnosti, ktoré chce anotovať, a to pomocou klikania pravým tlačidlom myši na atribút jedného objektu (produktu) a výberom typu atribútu.



Obrázok 1 Anotovanie zoznamu kávovarov pomocou scraper-u Connotate

V ďalšom kroku scraper identifikuje všetky podobné objekty na danej stránke a farebne ich zvýrazní. Scraper Connotate má funkcionalitu prehliadania anotovaných objektov, ktorou sa tiež dá ľahko overiť, či sa podarilo správne anotovať potrebné objekty. Zvýraznenie a prehliadanie anotovaných objektov, v našom prípade kávovarov, je zobrazené na nasledujúcom obrázku.



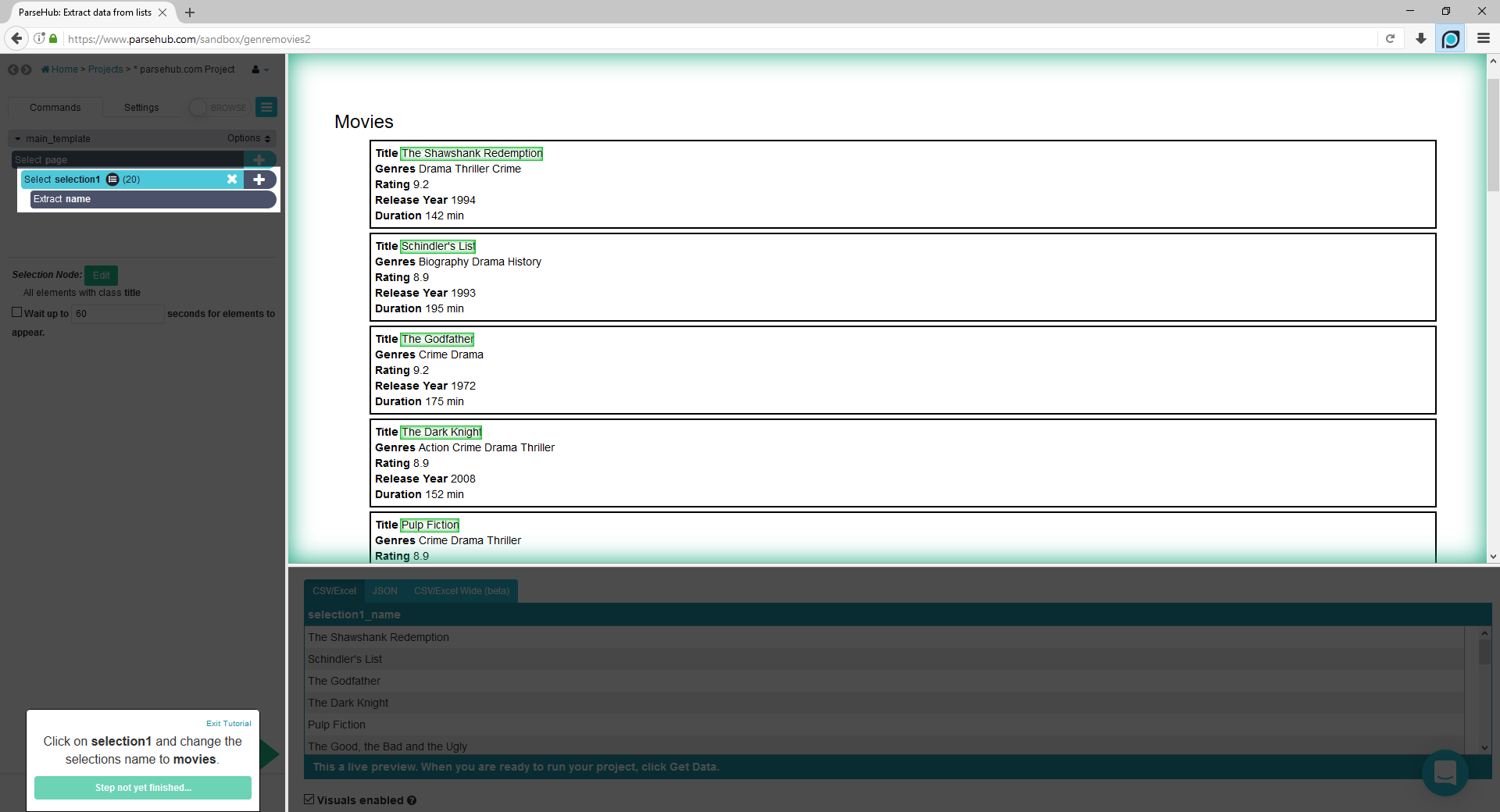
Obrázok 2 Zobrazenie anotovaných objektov scraper-u Connotate

Ak nastane prípad, keď nechceme objekty, ktoré boli automaticky označené, tak je potrebné kliknúť pravým tlačidlom myši na nechcený objekt a vybrať akciu „Ignore“. Následne sa objekt odstráni a nebude extrahovaný.

Vďaka mnohým vopred uloženým schémam, sa týmto spôsobom anotovania dá ušetriť čas. Často sa však stáva, že vo vopred uložených schémach je príliš veľa atribútov, ktoré nie sú potrebné, a preto je potrebné následne mazať takéto atribúty. Taktiež nie je možné vytvárať stromové štruktúry extrahovaných dát ako v iných scraper-och. Kladnou stránkou tohto prístupu je aj možnosť zdieľania schém s inými používateľmi.

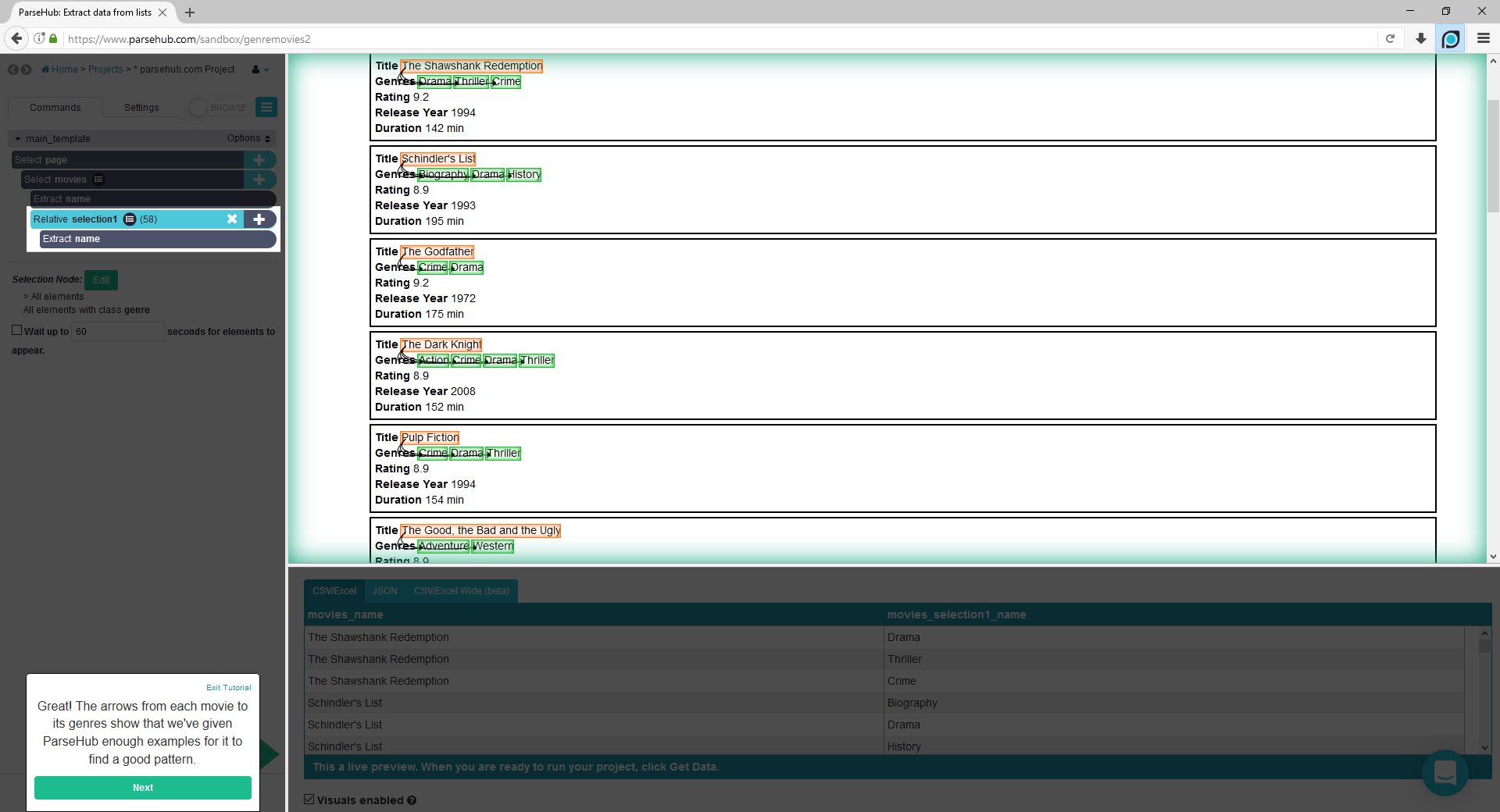
## ParseHub

Anotovanie viacerých objektov v ParseHub je realizované pomocou výberu atribútu viacerých objektov, pričom k týmto atribútom sa vytvoria ešte tzv. relatívne výbery. Na nasledujúcich obrázkoch je zobrazené, ako sa používateľ snaží extrahovať filmy zo zoznamu filmov. V prvom kroku si používateľ vyberie atribút, ktorý má každý film. V príklade na obrázku bol vybraný názov filmu, a to takým spôsobom, že používateľ klikol na názvy dvoch rôznych filmov. ParseHub následne zistil, že sa na stránke nachádza ešte viac podobných objektov (názov filmu) a automaticky označil ďalšie názvy filmov.



Obrázok 3 Označenie názvov filmov pre relatívny výber scraper-u ParseHub

Ďalej používateľ vytvorí relatívny výber ďalšieho atribútu k hlavnému atribútu pre jeden z filmov, čo v našom prípade znamená, že k vybranému názvu filmu chceme priradiť žáner. Nasledovne sa na webovej stránke zobrazia šípky smerujúce od názvu filmu k prislúchajúcemu žánru. Tento relatívny výber sa uloží do schémy.



Obrázok 4 Zobrazenie priradených objektov k názvom filmov pre scraper ParseHub

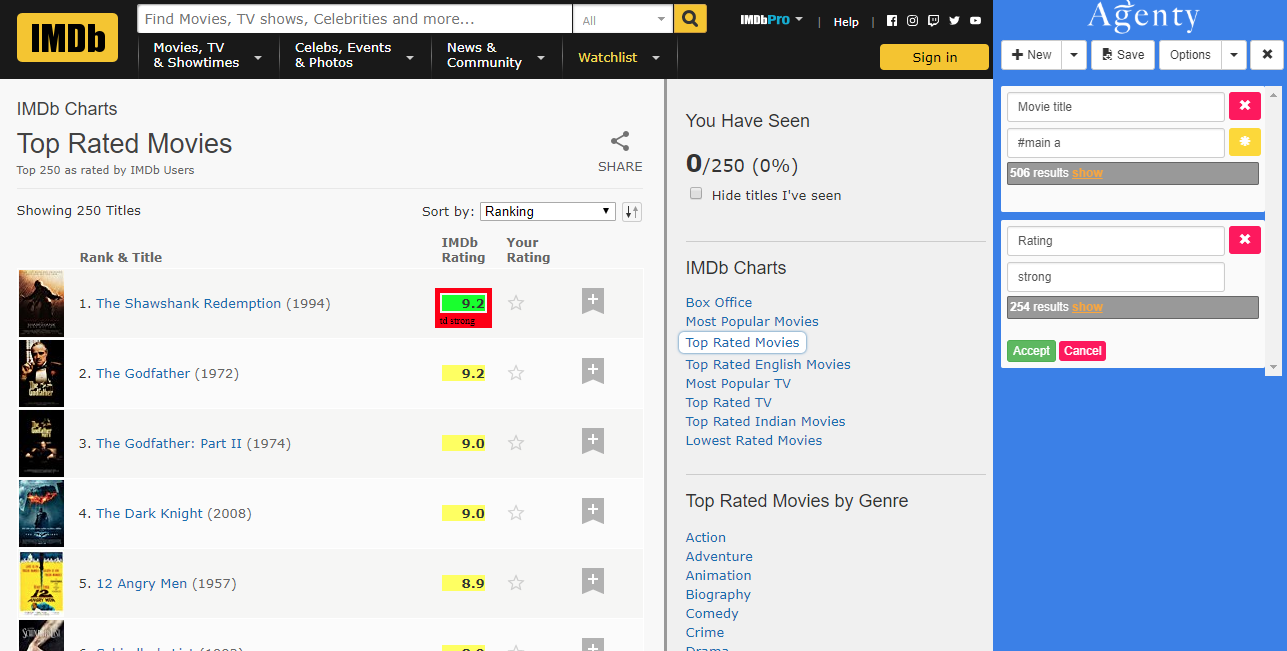
Vytvorenú schému (template), ktorú by sme mohli nazvať „zoznam filmov“, je možné znova použiť na inej webovej stránke, ale je potrebné zmeniť adresovacie hodnoty v schéme.

Tento prístup anotovania umožňuje priradiť objektu iný objekt alebo zoznam objektov. Extrahované dáta vytvárajú stromovú štruktúru, čo je vhodné pre rôzne analýzy dát. Scraper ParseHub však pracoval na zložitejších stránkach veľmi pomaly, čo bolo jeho najväčšou zápornou stránkou.

**Scraper-e fungujúce na podobnom princípe anotovania:** Content Grabber, Web Scraper.

## Agenty

V scraper-i Agenty je schéma pre viaceré objekty vytváraná postupne počas anotovania. Cieľom je vytvoriť zoznamy, v ktorých každý prvok prislúcha jednému z anotovaných objektov, pričom jeden zoznam predstavuje jeden atribút objektu. Na nasledujúcom obrázku znázorňujeme anotovanie filmov z [www.imdb.com](http://www.imdb.com), pričom chceme získať názov filmu a priradiť mu hodnotenie. Vytvoríme si teda dva zoznamy, kde prvý bude obsahovať názvy filmov a druhý hodnotenia týchto filmov. Označovanie jednotlivých atribútov je vyriešené poloautomaticky, kde používateľovi stačí kliknúť na jeden element, ako napríklad názov filmu, a následne sa automaticky označia všetky názvy filmov. Ak boli označené aj elementy, ktoré nie sú žiadúce, tak stačí znova kliknúť na takéto elementy a týmto spôsobom ich odstrániť zo zoznamu.



Obrázok 5 Vzhľad scraper-a Agenty

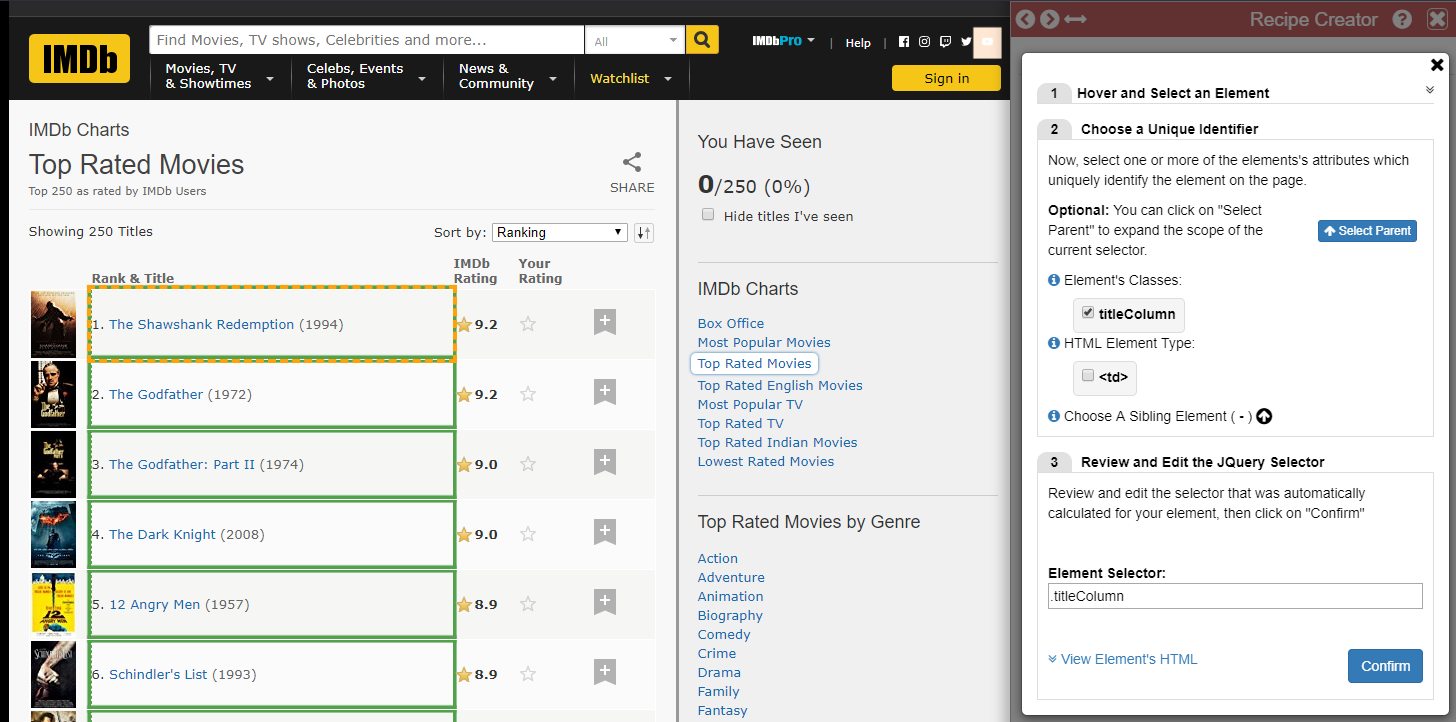
Takto vytvorené zoznamy sú následne spojené do jednotlivých objektov vo webovej aplikácii určenej na extrahovanie a plánovanie extrahovania.

Týmto prístupom anotovania je možné extrahovať dáta rýchlo a jednoducho. Veľkým nedostatkom je nefunkčnosť upravovania selekcie hodnôt v prípade, že nejaké hodnoty chýbali v selekcii, alebo sa tam vyskytovali také, ktoré sme nechceli. Po úprave selekcie sa často stáva, že buď z nej odstránime aj hodnoty, ktoré sme chceli mať v selekcii, alebo pridáme hodnoty, ktoré sme v nej nechceli mať. Ďalším nedostatkom je, že chýba možnosť extrahovania dát do stromovej štruktúry.

**Scraper-e fungujúce na podobnom princípe anotovania:** GetData.IO, Visual Web Ripper.

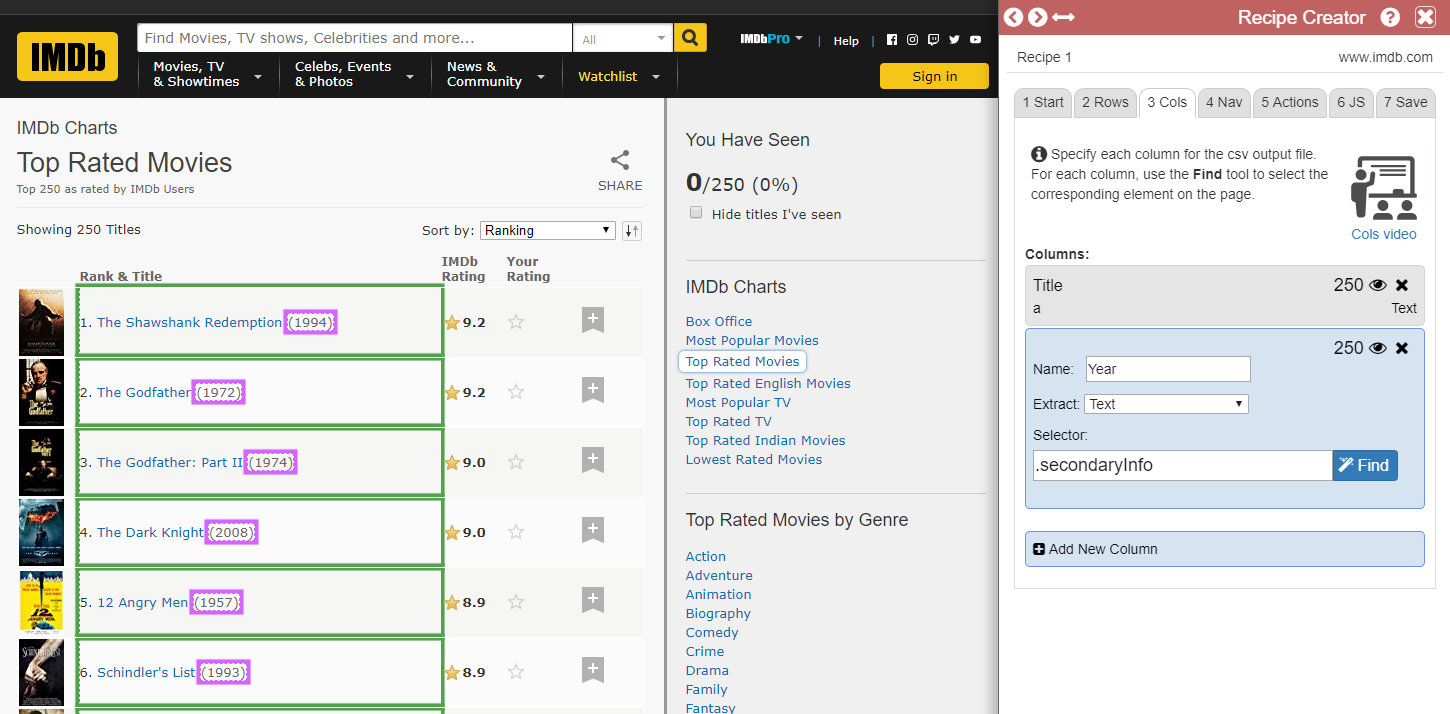
## Data Miner

So scraper-om Data miner si používateľ vytvorí novú schému (nový recept) a ako typ stránky si vyberie List page (zoznamová stránka). Následne scraper žiada, aby používateľ prešiel kurzorom myši nad jeden objekt zo zoznamu objektov a klikol na klávesu SHIFT. Týmto sa označí vybraný objekt a scraper požaduje od používateľa, aby vybral atribút HTML elementu, ktorý má označený objekt. Mal by byť vybraný taký atribút, aby sa následne označili všetky objekty, ktoré používateľ chce anotovať. Tento proces je zobrazený na nasledujúcom obrázku.



Obrázok 6 Anotovanie objektov zo zoznamu pomocou scraper-a Data Miner

Ďalej je potrebné pre každý objekt vybrať sériu atribútov. Ako príklad si pre každý z vybraných filmov vyberieme názov filmu a rok premiéry rovnakým spôsobom, ako sme vyberali celé objekty na predchádzajúcom obrázku. Na nasledujúcom obrázku je vidieť označené roky premiéry pre každý z vybraných filmov, pričom scraper zobrazuje aj ich počet.



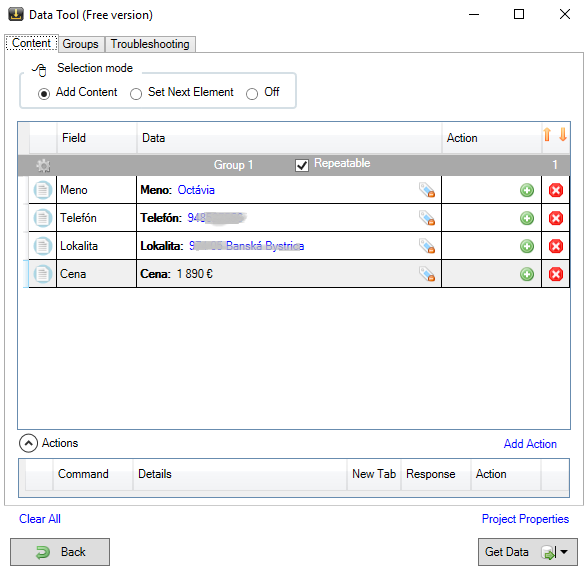
Obrázok 7 Anotovanie atribútov pre objekty

Tento spôsob anotovania je používateľsky prívetivý a jednoduchý. Používateľa sprevádza scraper v každom kroku, a popisuje mu, čo má robiť. Na uniformných stránkach funguje veľmi dobre, avšak na zložitejších stránkach nie je postačujúci. Ďalšou nevýhodou je neschopnosť uložiť dáta do stromovej štruktúry a úprava selekcie, ak nejaký atribút HTML elementu nevyhovuje potrebám anotovania zoznamu.

**Scraper-e fungujúce na podobnom princípe anotovania:** Easy Web Extract, Grepsr, Scrape.it, WebHarvy.

## Data Toolbar

Tento scraper funguje veľmi podobne ako Data Miner, ale pri anotovaní stačí označiť len atribúty jedného objektu. Napríklad pri anotovaní inzerátov aut zo stránky [www.bazos.sk](http://www.bazos.sk) si používateľ klikne na telefónne číslo predávajúceho, značku, lokalitu a cenu auta. Potom sa tieto štyri atribúty uložia ako Field1, Field2, atď. Názvy týchto atribútov je možné zmeniť, ako to vidíme na nasledujúcom obrázku. Zaškrtnutím políčka „Repaeatable“ a kliknutím na tlačidlo „Get Data“ sa automaticky identifikujú všetky objekty na stránke a následne sa extrahujú.



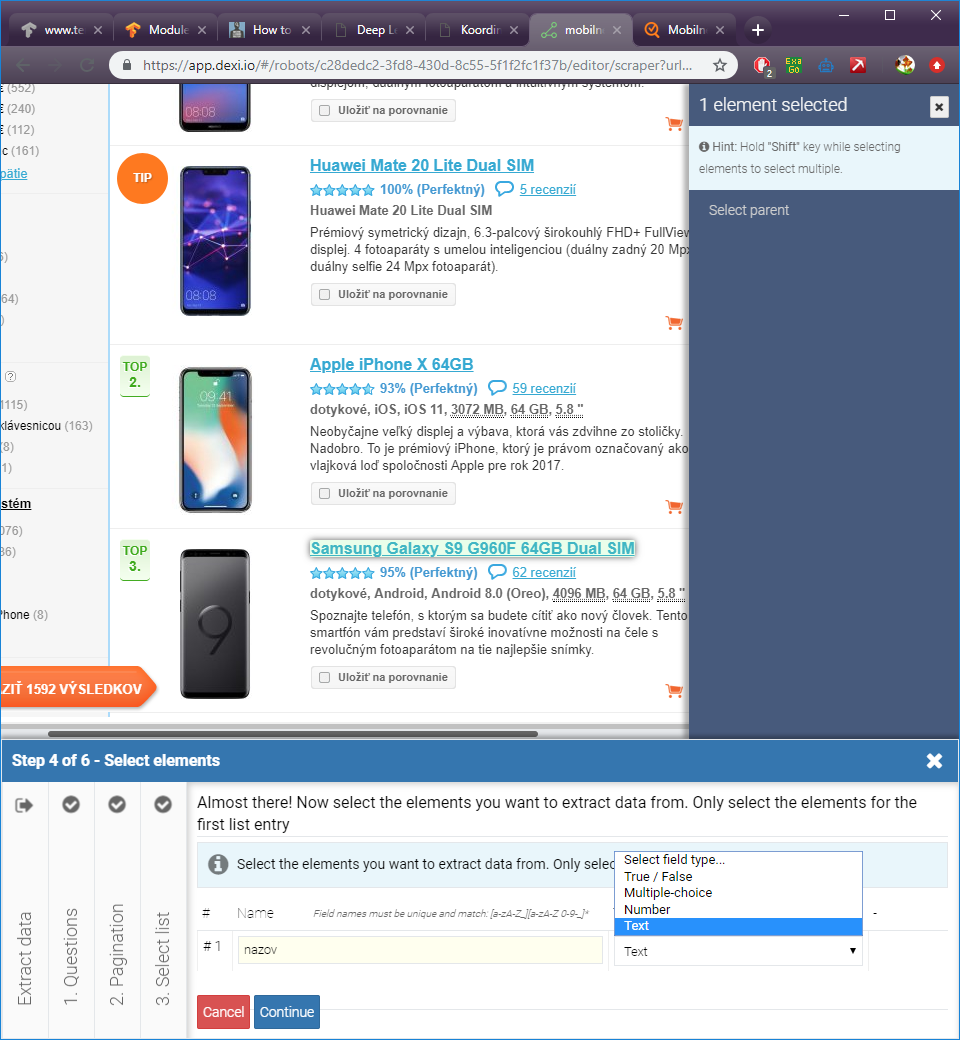
Obrázok 8 Anotovanie zoznamu objektou scraper-om Data Toolbar

Výhodou tohto riešenia je používateľsky prívetivé prostredie a rýchlosť anotovania, keďže je potrebné anotovať len jeden objekt a netreba anotovať ďalšie objekty zo zoznamu. Jednou z nevýhod je neschopnosť uložiť extrahované dáta do stromovej štruktúry. Ďalšou nevýhodou je neschopnosť upravovať nesprávne označené atribúty. Používateľ musí poznať syntax XPath-ov, aby vedel upraviť takéto atribúty.

## Dexi.io

Anotovanie scraper-om Dexi.io je robené pomocou otázok, na ktoré má používateľ odpovedať. Prvou sadou otázok sa tento scraper pýta, či používateľ chce anotovať zoznam. Ak áno, tak scraper zisťuje, či dáta sú rozmiestnené na viacerých stránkach, a či zoznam objektov obsahuje odkaz na detailovú stránku objektu. Podľa toho, ako používateľ odpovedá, tak sa postupne od neho žiada, aby anotoval potrebné elementy na stránke. Samotné anotovanie objektov prebieha formou anotovania jednotlivých atribútov, z ktorých sa vytvorí tabuľka, kde každý riadok reprezentuje jeden objekt.

Na nasledujúcom obrázku je znázornené, ako sú anotované mobily zo zoznamu mobilov na stránke [www.heureka.sk](http://www.heureka.sk). Kliknutím na názov jedného mobilu je používateľovi navrhnuté, či si praje označiť zvyšné podobné elementy, ktoré boli nájdené scraper-om na tejto stránke. Ďalej sa žiada, aby bolo vybrané, čo sa má extrahovať z anotovaných elementov, a ako sa bude nazývať druh týchto elementov. V tomto prípade sme vybrali ako druh vybraného elementu „text“ a pomenovali sme ho „nazov“. Postupne je možné anotovať aj ďalšie atribúty mobilov ako cena, popis či hodnotenie.



Obrázok 9 Anotovanie jednotlivých atribútov objektov scraper-om Dexi.io

Tento spôsob anotovania je používateľsky veľmi prívetivý a pre používateľa aj časovo výhodný. Nie je potrebné študovať tutoriály, no napriek tomu používateľ vie v každom kroku, čo má robiť. Zápornou stránkou je neschopnosť extrahovať dáta do stromových štruktúr, či upravovať selekcie pri nesprávnom označení.

## FMiner

Anotovanie so scraper-om FMiner je tvorené pomocou sledu akcií, podľa ktorého je scraper navigovaný po webovej stránke a vykonáva extrakciu. Tento sled akcií sa skladá z rôznych uzlov, ktoré reprezentujú rôzne akcie. Vytvára ho používateľ, pričom prvý uzol je vždy „goto“. V tomto uzle je potrebné špecifikovať URL adresu, na ktorej sa bude vykonávať extrakcia dát. Používateľ po kliknutí na tlačidlo „Run“ spustí sled akcií, čiže v tomto prípade sa nám len načíta stránka. Potom si používateľ vytvorí uzol „scrape page“, ktorý vraví o tom, že na tejto stránke sa budú extrahovať dáta. Pre tento uzol je potrebné vytvoriť tabuľku a nastaviť jej názvy stĺpcov.

Na nasledujúcom obrázku je zobrazené, ako používateľ chce extrahovať názvy filmov a ich hodnotenia na stránke [www.imdb.com](http://www.imdb.com) . Vytvorili sme teda tabuľku s názvom „Filmy“ a stĺpce „Názov“ a „Hodnotenie“. Pre anotovanie viacerých elementov obsahujúcich hodnotenie používateľ najprv klikne na jeden element. Následne klikne na tlačidlo „group selection“, a potom klikne na druhý element. Tak sa vygeneruje XPath, ktorý označuje aj všetky ostatné hodnotenia.



Obrázok 10 Anotovanie zoznamu filmov scraper-om FMiner

Výhodou tohto riešenia je možnosť vytvoriť viacero tabuliek (zoznamov) rôznych objektov, ktoré nemusia so sebou súvisieť. Postup anotovania je veľmi intuitívny a dáva používateľovi mnoho možností. Nevýhodou je, že extrahované dáta sa neukladajú do stromovej štruktúry a scraper neumožňuje upravovať nesprávne označené atribúty bez znalosti syntaxe jazyka XPath. Scraper-i FMiner umožňuje písať aj regulárne výrazy, avšak negeneruje ich, čo je ďalšou nevýhodou pre používateľa.

**Scraper-e fungujúce na podobnom princípe anotovania:** Helium Scraper, Mozenda, Web Sundew, Web Content Extractor.

# Anotovanie detailovej stránky objektu

Ako detailovú stránku budeme chápať stránku, na ktorej sú zobrazené detaily jedného objektu. Na tejto stránke sa môžu nachádzať rôzne tabuľky prislúchajúce pre tento objekt, zoznam obrázkov, názvy atribútov s priradenými hodnotami a ďalšie charakteristiky objektu. V tejto kapitole si ukážeme anotovanie detailovej stránky vo viacerých scraper-och, rozdelených do kategórií podľa podobnosti.

## Connotate

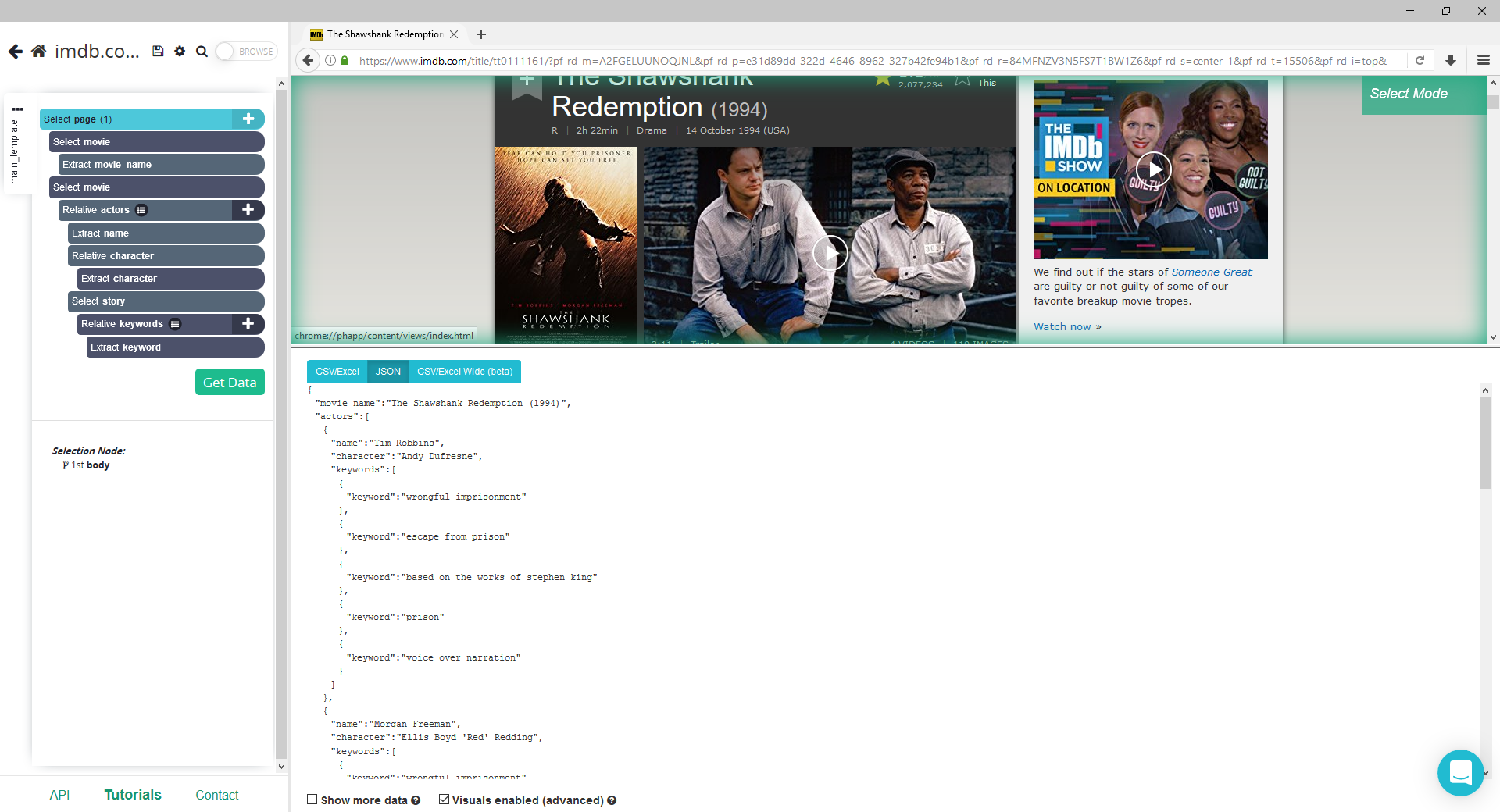
Rovnako ako pri anotovaní viacerých objektov zobrazených na jednej webovej stránke, tak aj pri anotovaní jedného objektu zobrazeného na jednej webovej stránke bolo potrebné vybrať schému, podľa ktorej musel používateľ vykonať anotáciu. Vo veľkej väčšine prípadov takáto schéma nebola dostupná a používateľ si ju musel vytvoriť. Veľké pozitívum tohto scraper-a – použitie vopred uložených schém, bolo v tomto prípade zanedbateľné.

## ParseHub

Spôsob anotovania týmto scraper-om funguje na princípe vytvorenia ľubovoľnej selekcie a k tejto selekcii sa priradí nejaká akcia. Tieto akcie sú:

* extrakcia vybranej selekcie,
* vytvorenie relatívnej selekcie k vybranej selekcii,
* kliknutie na danú selekciu,
* vyplnenie formulára,
* presunutie myši nad túto selekciu,
* rolovanie myši,
* vytvorenie zoznamu.

Na nasledujúcom obrázku je zobrazený postup anotácie konkrétneho filmu z webovej stránky [www.imdb.com](http://www.imdb.com), kde cieľom bolo extrahovať názov filmu, zoznam hercov, pre každého herca meno jeho role vo filme a ku každému hercovi zoznam kľúčových slov filmu.



Obrázok 11 Zobrazenie stromovej štruktúry anotovaných dát v scraper-i ParseHub

Výstup po extrakcii bol uložený do stromovej štruktúry, čo je veľkým pozitívom, a množina dostupných akcií je postačujúca pre väčšinu použití. Postup anotovania bol zdĺhavý a pre používateľa náročný. Na to aby používateľ vedel vytvoriť takýto jednoduchý príklad, potreboval vytvoriť veľké množstvo akcií a selekcií. Týmto prístupom bolo komplikované vytvoriť vzťahy medzi objektmi, ktoré používateľ chcel extrahovať.

**Scraper-e fungujúce na podobnom princípe anotovania:** Content Grabber, Web Scraper.

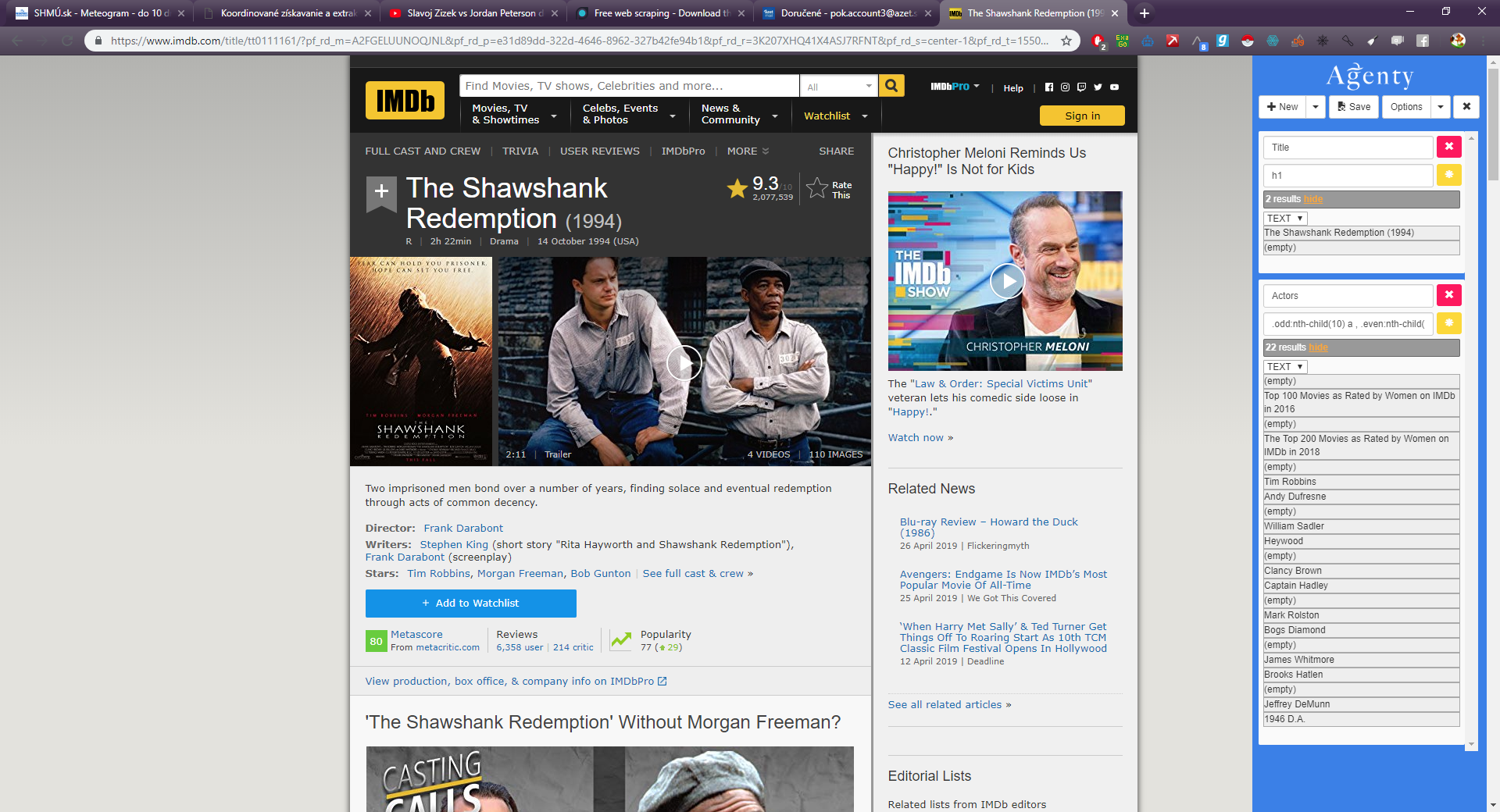
Postup anotovania jedného objektu v scraper-i Content Grabber bol pre používateľa ešte náročnejší, kvôli zastaranému vzhľadu a ešte väčšiemu počtu akcií používateľa potrebných na vykonanie anotovania.

Web Scraper bol pre nového používateľa používateľsky neprívetivý, pretože neobsahoval žiaden tutoriál alebo popis jeho funkcionalít. Nebolo hneď jasné, čo mal používateľ urobiť na vykonanie extrakcie a k výsledkom sa dostal až po zdĺhavom skúmaní tohto scraper-a. Taktiež obsahoval menšie množstvo funkcionalít ako ParseHub.

## Agenty

Anotovanie jedného objektu zobrazeného na jednej stránke v tomto scraper-i sa nelíši od anotovania zoznamu objektov. Kkeď sme chceli anotovať názov konkrétneho filmu a zoznamu hercov na stránke [www.imdb.com](http://www.imdb.com), tak sa nám vyskytli chyby, ktoré nebolo možné napraviť. Nebolo možné anotovať len názov filmu, do extrakcie nám scraper uložil aj rok vydania a ešte prázdny objekt. Navyše Agenty nedokázal ani vygenerovať CSS výber pre zoznam hercov, ktorý by ich vedel všetkých adresovať. Taktiež bolo do zoznamu hercov uložených mnoho prázdnych objektov. Nebola iná možnosť ako len manuálne upraviť CSS výbery, čo je pre používateľov, ktorí nepoznajú túto syntax, náročné.

Na nasledujúcom obrázku je zobrazená neúspešná anotácia názvu filmu a zoznamu hercov. Prázdne objekty v tomto scraper-i sa nazývajú „(empty)“.



Obrázok 12 Neúspešné anotovanie detailovej stránky filmu pomocou scraper-a Agenty

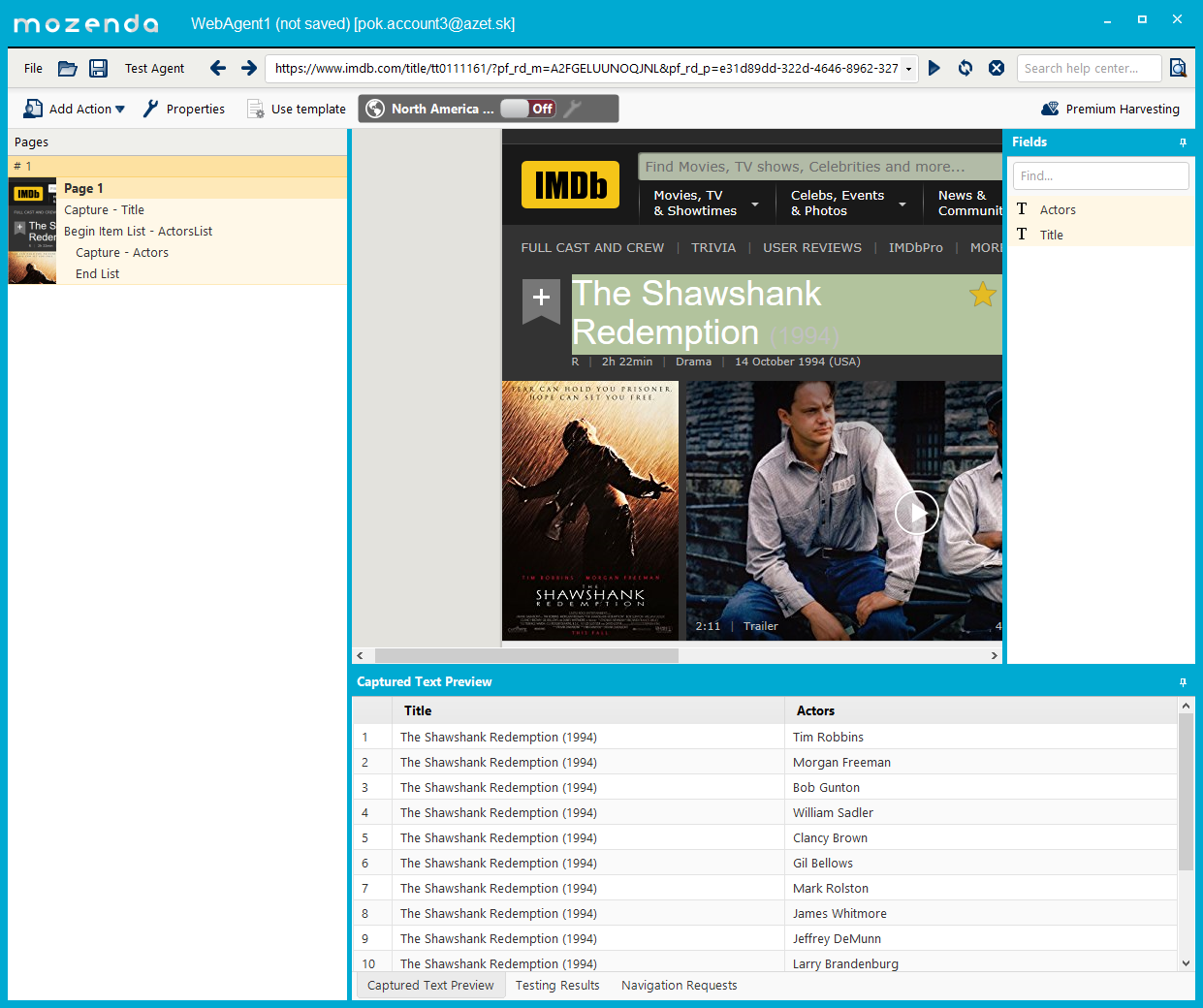
**Scraper-e fungujúce na podobnom princípe anotovania:** GetData.IO, Visual Web Ripper.

Scraper-e GetData.IO a Visual Web Ripper ktoré pracujú podobne ako scraper Agenty, dokázali anotovať názov filmu a zoznam hercov bez problémov a extrakcia taktiež prebehla úspešne.

## Mozenda

Postup anotovania v tomto scraper-i pozostáva z rôznych akcií. Tieto akcie sú: anotovanie zoznamu, páru meno a hodnota, tabuľky, obrázku a bloku textu. Ďalej je možné vyplniť vstup, kliknúť na objekt, vymazať cookies, zachytiť obsah obrazovky (tzv. screenshot), obnoviť stránku, počkať niekoľko sekúnd, spustiť JavaScript a stiahnuť súbor.

Na nasledujúcom obrázku je zobrazené úspešné anotovanie názvu filmu a zoznamu hercov. V dolnej časti obrázka je zobrazená tabuľka s dátami, ktoré boli anotované, pričom výstup po extrakcii dát vyzerá rovnako. Nie je možné uložiť dáta do stromovej štruktúry, a teda často sa nám opakujú rovnaké dáta. V tomto príklade sa nám opakuje názov filmu, ktorý je priradený ku každému hercovi. V stromovej štruktúre by sme mohli k názvu filmu priradiť zoznam hercov a znížiť redundanciu. Výhody tohto scraper-a je veľký počet akcií, ktoré umožnia používateľovi vykonať ľubovoľnú extrakciu.



Obrázok 13 Zobrazenie tabuľky v scraper-I Mozenda

**Scraper-e fungujúce na podobnom princípe anotovania:** FMiner,Helium Scraper, Web Sundew, Web Content Extractor.

Scraper FMiner dokázal anotovať ako zoznam hercov, tak aj názov filmu, ale s extrakciou dát mal problémy. Zoznam hercov, ktorý sa mal uložiť do tabuľky, bol uložený nesprávne. Namiesto toho, aby v každom riadku tabuľky bolo jedno meno herca, tak do jednej bunky tabuľky boli vložené všetky mená hercov.

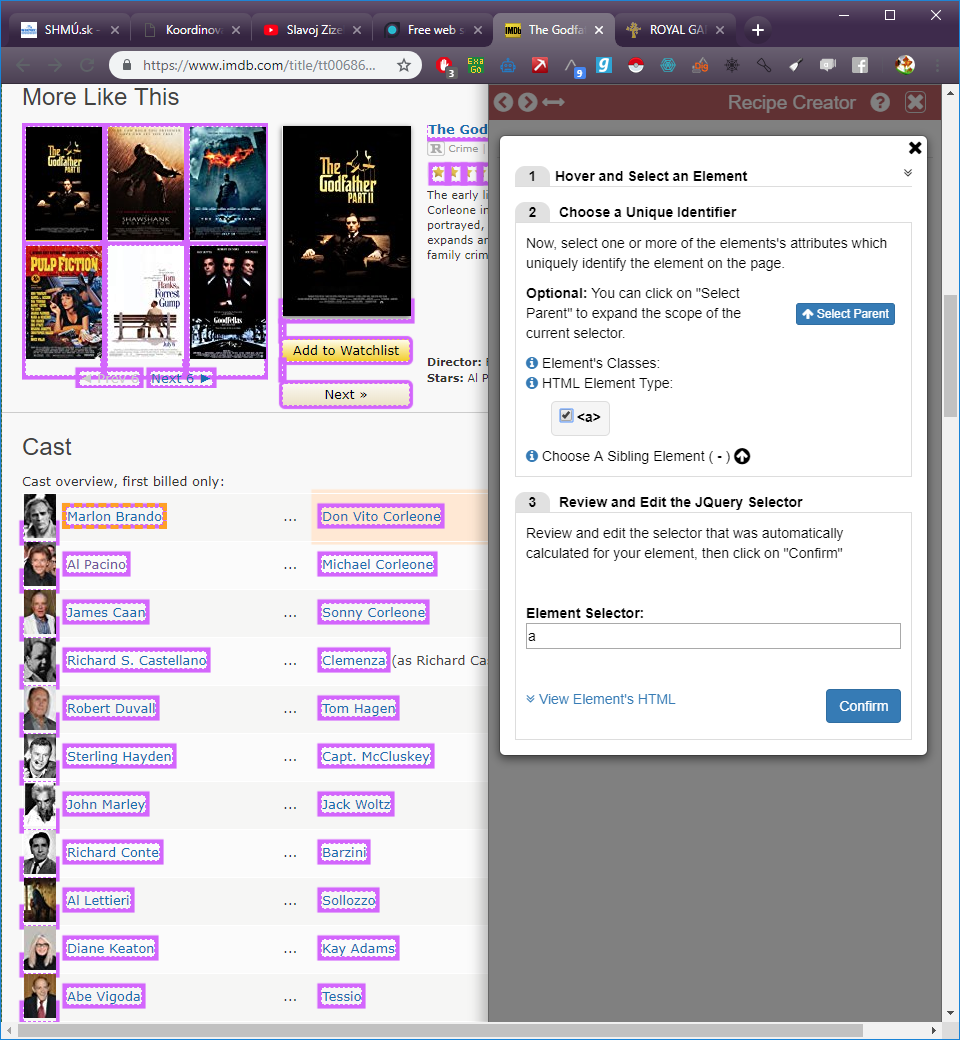
Helium Scraper-om sa podarilo anotovať a extrahovať aj názov filmu aj zoznam hercov. Taktiež bolo možné upravovať selekciu dát, ak nejaký element v tejto selekcii chýbal. Výsledky boli rovnako uložené do tabuľky, ktorej štruktúru navrhol samotný scraper, ale poradie stĺpcov a čo sa malo z anotovaných elementov sa dalo upravovať. Z elementov bolo možné extrahovať: text, OuterHTML, odkaz, URL adresu a jedno-riadkový vnútorný text (tzv. inner text).

Scraper Web Sundew a Web Content Extractor boli používateľsky neprívetivé a neobsahovali žiadne ďalšie užitočné funkcionality.

## Data Miner

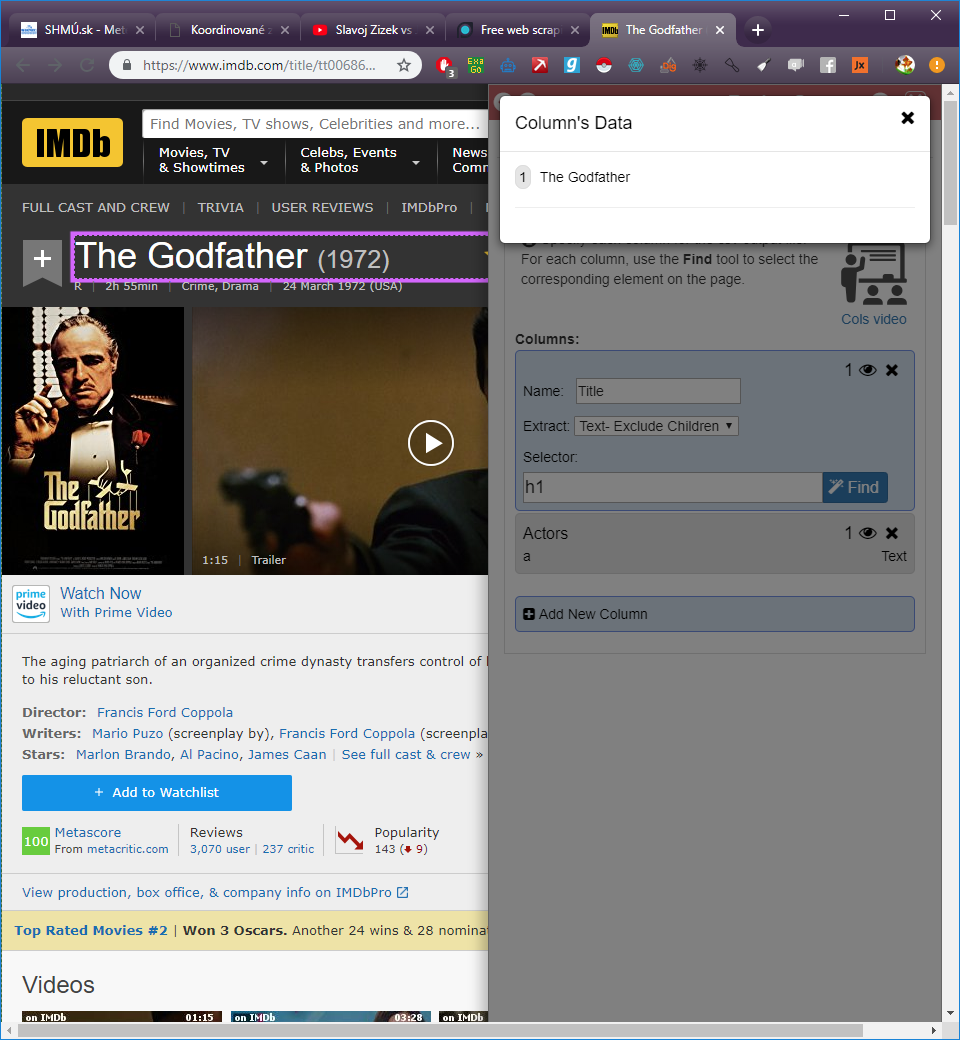
V prvom kroku anotovania jedného objektu je potrebné zvoliť ako typ stránky detailovú stránku. Následne scraper žiada, aby používateľ prešiel kurzorom myši na HTML element, ktorý chce anotovať, a klikol na klávesu SHIFT. Týmto sa označí vybraný element a scraper požaduje od používateľa, aby vybral atribút tohto elementu. Ak vybraný HTML element nemá žiaden atribút, tak používateľovi nie je daná žiadna možnosť na výber a musí si vybrať len názov tohto elementu. Toto je veľký nedostatok tohto scraper-a, kvôli ktorému sme nedokázali anotovať zoznam hercov pre vybraný film.

Na nasledujúcom obrázku je zobrazené anotovanie zoznamu hercov, kde nám bolo navrhnuté vybrať si len názov elementu „<a>“. Takýto element sa však vyskytoval aj na iných miestach na webovej stránke, nie len pri zozname hercov. Žiadne ďalšie úpravy na spresnenie selekcie neboli k dispozícii.



Obrázok 14 Neúspešné anotovanie zoznamu atribútov pre jeden objekt (film) v scraper-i Data Miner

Pri výbere názvu filmu sme využili možnosť extrahovania textu z elementu okrem elementov, ktoré boli v ňom obsiahnuté. Táto funkcionalita sa nazýva „Extract Text - Exclude Children“ a umožnila nám vynechať rok premiéry filmu, ktorý sa nachádzal v elemente spolu s názvom filmu. Táto funkcionalita je zobrazená na nasledujúcom obrázku.



Obrázok 15 Zobrazenie funkcionality Text-Exclude Children v scraper-i Data Miner

**Scraper-e fungujúce na podobnom princípe anotovania:** Easy Web Extract, Grepsr, Scrape.it, WebHarvy, Dexi.IO.

Pomocou scraper-ov Easy Web Extract a Grepsr sa nám podarilo extrahovať názov filmu aj zoznam hercov bez problémov. Easy Web Extract mal však používateľsky neprívetivé prostredie a Grepsr nemal žiadne ďalšie funkcionality ako len anotovanie zoznamu alebo anotovanie jedného elementu.

Scraper-om Scrape.it sme dokázali anotovať názov filmu aj zoznam hercov, avšak po anotovaní nie je možné nič iné vykonať. Tým pádom sa nedá nič extrahovať a celý scraper dokáže len anotovať objekty.

Scraper WebHarvy nedokázal anotovať zoznam hercov, pretože jeho funkcionalita „Capture more content“ fungovala na princípe vynárania sa od anotovaného elementu o úroveň vyššie a zachycovania všetkých elementov nachádzajúcich sa pod aktuálnym elementom. V našom prípade boli anotované aj role hercov, čo sme nechceli. Tento scraper však mal zaujímavú funkcionalitu – výber z dropdown menu.

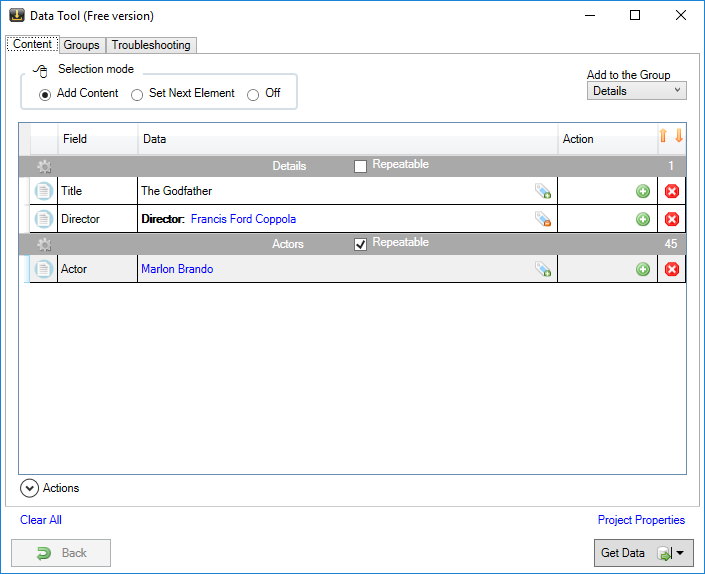
Zoznam hercov nedokázal anotovať ani scraper Dexi.IO, pretože do zoznamu vždy pridal aj role hercov. Anotovanie bolo veľmi podobné scraper-u Data Miner.

## Data Toolbar

Anotovanie detailovej stránky filmu scraper-om Data Toolbar sa líši od anotovania zoznamu objektov iba začiarknutím políčka „repeatable“ pri niektorých skupinách atribútov. Pri anotovaní názvu filmu a režiséra sme vytvorili skupinu „Details“.V tejto skupine sme políčko „repeatable“ nezačiarkli, a teda je vhodné pridávať do tejto skupiny len jednoduché atribúty - názov atribútu a hodnota. Pri anotovaní režiséra sme využili funkcionalitu „Add prompt to element“, ktorá umožní anotovať typ objektu. Nebolo teda potrebné vypisovať názov stĺpca, stačilo kliknúť na text „Director“, ktorý sa nachádzal na webovej stránke. Táto funkcionalita je používateľsky prívetivá a pri anotovaní viacerých objektov šetrí používateľovi čas.

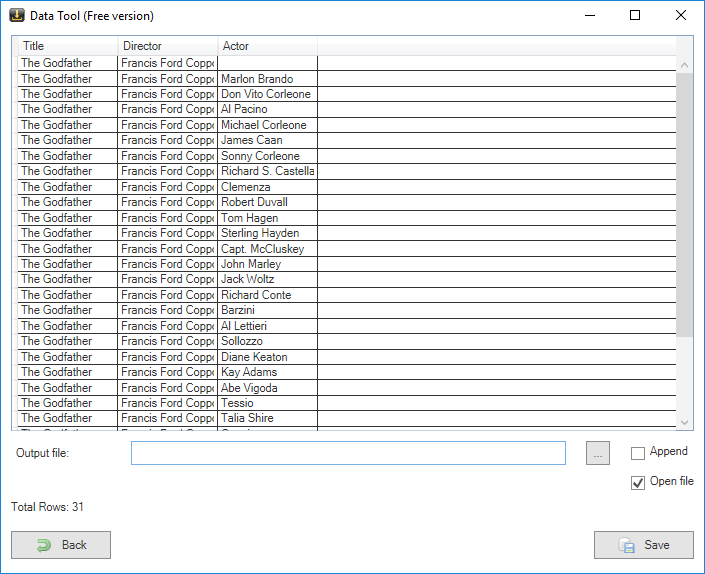
Pri anotovaní zoznamu hercov už bolo potrebné vytvoriť si ďalšiu skupinu, ktorú sme pomenovali „Actors“ a začiarknuť políčko „repeatable“, ktoré vraví o tom, že sa na danej stránke bude vyskytovať viac podobných objektov. Ďalej je potrebné kliknúť na meno jedného z hercov, scraper identifikuje ďalšie podobné objekty automaticky a vygeneruje XPath, ktorý ich adresuje. Anotovanie zoznamu hercov nebolo úspešné, pretože vygenerovaný XPath adresoval aj role pre každého herca, čo sme nechceli. XPath bolo možné prepísať, avšak my nepredpokladáme, že každý používateľ pozná syntax jazyka XPath.

Na nasledujúcom obrázku je zobrazené anotovanie názvu filmu, režiséra a zoznamu hercov. V skupine „Actors“ je zobrazený počet 45, pričom počet hercov bol 15 a po extrahovaní dát sme dostali 30 riadkov. Zobrazený počet identifikovaných hercov bol teda nepravdivý.



Obrázok 16 Anotovanie atribútov konkrétneho filmu v scraper-i Data Toolbar

Na nasledujúcom obrázku sú zobrazené dáta po extrakcii uložené do tabuľky. Tieto dáta nie sú uložené v stromovej štruktúre a veľmi často sa opakuje názov filmu a meno režiséra.



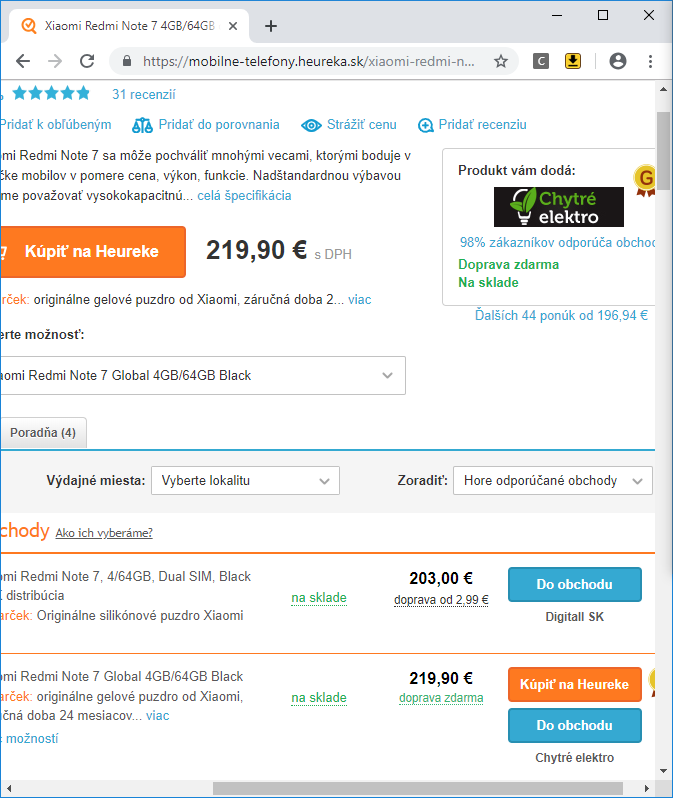
Obrázok 17 Dáta zobrazené po extrakcii v scraper-i Data Toolbar

# Anotovanie jedného objektu zobrazeného na viacerých webových stránkach

Pri anotovaní jedného objektu často nastáva situácia, že detaily o tomto objekte sú zobrazené na viacerých detailových stránkach. Ďalej môže nastať situácia, keď objekt, ktorý chceme anotovať, sa nachádza v zozname objektov, pričom tento objekt má na takejto zoznamovej stránke iba malú časť atribútov. Po kliknutí na tento objekt sa zobrazí jeho detailová stránka, na ktorej sa nachádza väčšia časť jeho atribútov, napríklad nejaká tabuľka, zoznam obrázkov alebo komentáre k tomuto objektu. V tejto kapitole si ukážeme, ako sa dajú takéto objekty čo najľahšie anotovať.

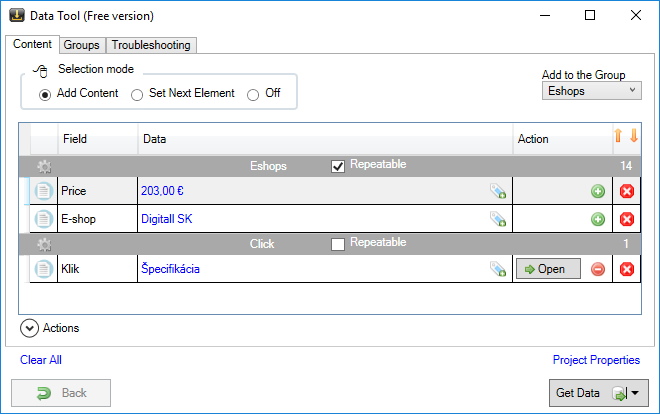
Všetky scraper-e, ktoré dokázali anotovať takýto objekt, fungovali na princípe prekliknutia na inú stránku a anotovania na tejto novej stránke. Scraper-e podporujúce anotovanie jedného objektu zobrazeného na viacerých stránkach sú: Connotate, ParseHub, Data Miner, Data Toolbar, Dexi.IO, Easy Web Extract, FMiner, Helium Scraper, Import.IO, Mozenda, Visual Web Ripper, Web Scraper, Web Sundew, Scrape.it, Web Content Extractor a Web Harvy.

Na nasledujúcom obrázku je zobrazená webová stránka smartfónu na portáli [www.heureka.sk](http://www.heureka.sk). Na tejto stránke budeme chcieť anotovať zoznam obchodov a cien v týchto obchodoch.



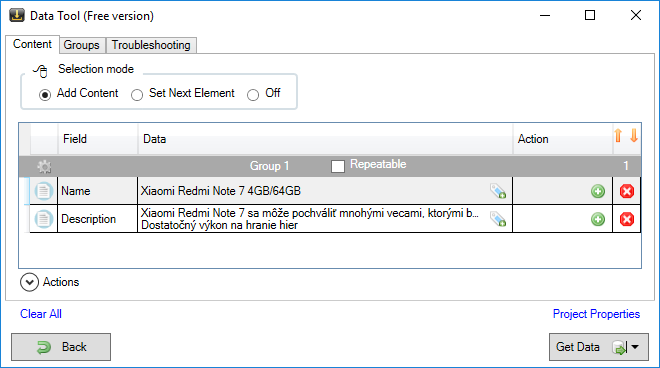
Obrázok 18 Anotovanie zoznamu obchodov a cien pre smartfón

Po anotovaní názvu obchodov a cien sa z tejto stránky dostaneme na stránku špecifikácie smartfónu a anotujeme tu názov a popis. Na nasledujúcom obrázku je zobrazený scraper Data Toolbar, v ktorom sme vytvorili kliknutie pomocou akcie „Open“. Používateľ potom klikne na tlačidlo „Open“ a do prehliadača sa mu načíta stránka špecifikácie.



Obrázok 19 Anotovanie objektu zobrazeného na viacerých stránkach pomocou scraper-a Data Toolbar a akcie “klik”

Po otvorení stránky špecifikácie sa používateľovi zobrazí prázdny scraper, do ktorého pridá stĺpce názov smartfónu a popis. Na nasledujúcom obrázku je zobrazená anotácia stránky špecifikácie. Ak používateľ klikne na tlačidlo “Back”, tak scraper automaticky načíta predchádzajúcu stránku a zobrazí sa anotácia ako na predchádzajúcom obrázku.



Obrázok 20 Anotovanie stránky špecifikácie pre smartfón

Tento spôsob je pre používateľa dostatočne intuitívny a nie je potrebné nastavovať veľké množstvo parametrov. Rovnako by sme postupovali aj v prípade, keby sa objekt nachádzal v zozname objektov. Časť jeho atribútov by sme získali zo zoznamu a časť z jeho detailovej stránky.

# Prechádzanie stránok

Pri extrahovaní viacerých objektov z webového portálu je väčšinou potrebné prechádzať viacero stránok, na ktorých sa každý z týchto objektov nachádza. Všetky scraper-e, ktoré sme testovali, fungovali tak, že na začiatku bolo potrebné nájsť zoznam objektov, cez ktorý sa dalo pomocou prekliknutia na každý z objektov dostať na prislúchajúcu detailovú stránku. Ak bol zoznam objektov rozložený na viacerých webových stránkach, tak bolo potrebné zadať typ stránkovania, tzv. „pagináciu“, pomocou ktorej scraper vedel prechádzať viacero stránok, a tým nájsť všetky objekty zoznamu. V prípade, keď sme chceli extrahovať viacero zoznamov objektov, tak bolo potrebné nájsť každý zoznam a priradiť k nemu typ stránkovania. Z toho vyplýva, že ak na webovom portáli neexistoval zoznam všetkých objektov, čiže objekty boli uložené do viacerých nesúvisiacich zoznamov, tak používateľ musel anotovať osobitne každý zoznam, aby vedel extrahovať všetky objekty z tohto portálu.

# Typy extrahovaných dát

V tejto kapitole si ukážeme, aké typy dát dokážu extrahovať scraper-e, ktoré sme testovali.

## Text

Všetky scraper-e dokázali extrahovať text z anotovaného elementu, ktorý sa v testovaných scraper-och nazýval text alebo „inner text“. Výsledkom po extrakcii bol text, ktorý obsahoval anotovaný element a text obsiahnutý vo všetkých potomkoch tohto elementu. Takéto texty vidí používateľ na webovej stránke a je schopný ich označiť myšou a skopírovať.

Scraper Data Miner bol schopný extrahovať text len anotovaného elementu. Texty jeho potomkov odignoroval a nenachádzali sa vo výstupe. V niektorých scraper-och bolo na výber extrahovanie samotných čísel z textu.

## Outer a inner HTML elementu

Ďalšími typmi dát na extrakciu boli tzv. outer a inner HTML anotovaného elementu. Výsledok po extrakcii outer HTML je celý HTML kód elementu a jeho detí. Po extrakcii inner HTML je výsledok podobný outer HTML, avšak vo výsledku už nie je obsiahnutá otváracia a uzatváracia značka, tzv. „tag“ elementu, ktorý sme anotovali.

## Atribúty HTML elementov a sťahovanie súborov

Väčšina scraper-ov dokázalo extrahovať len vopred vybrané hodnoty atribútov HTML elementov, ako napríklad „href“, v ktorom je uložená URL adresa, na ktorú tento element odkazuje, alebo „title“, čo je názov konkrétneho elementu. Niektoré scraper-e boli schopné extrahovať všetky typy atribútov a priradiť im ich hodnoty. Tieto atribúty môžu byť napríklad „href“, „title“, „class“ alebo „style“.

Pomocou atribútu „href“ a „src“ dokázali scraper-e stiahnuť súbory, na ktoré tieto odkazy smerovali. Týmto spôsobom bolo možné stiahnuť ľubovoľný súbor alebo obrázok.

Taktiež bolo možné extrahovať aj samotný názov značky elementu. Napríklad, ak sme mali značku „<img>”, tak výsledkom bol reťazec „IMG“.

## URL adresy

Ďalším typom dát na extrakciu bola URL adresa webovej stránky alebo rodičovská URL adresa. URL adresa webovej stránky je tá URL adresa, na ktorej sa nachádzajú extrahované dáta. Rodičovská URL adresa sa používa, ak je na webovej stránke použitá tzv. „iframe“ značka, v ktorej je zobrazený obsah inej webovej stránky. Z tejto webovej stránky nachádzajúcej sa v „iframe“ teda vieme získať URL adresu rodičovského HTML dokumentu, ktorý obsahuje daný „iframe“.

# Počiatočný stav Exaga

Verzia Exaga, z ktorej sme vychádzali, je špeciálne navrhnutá, aby dokázala anotovať a extrahovať stránky z internetových obchodov. Chceli by sme túto verziu Exaga prerobiť na univerzálny webový scraper, ktorý by dokázal extrahovať ľubovoľné objekty.

# Návrh riešenia

## Anotovanie ľubovoľného objektu

Exago-m je možné anotovať detailovú stránku produktu, pričom názvy atribútov sú fixné. Je možné anotovať cenu, názov, popis, identifikátor produktu a iné vlastnosti produktov v internetových obchodoch. Ďalej je možné anotovať zoznam názvu atribútu s priradenou hodnotou, pričom používateľovi stačí kliknúť na jeden z názvov atribútov. Ďalej sa automaticky identifikujú aj ďalšie názvy atribútov a k nim priradia prislúchajúce hodnoty. Nie je však možné vytvoriť názov ľubovoľného atribútu a priradiť k nemu hodnotu.

Navrhujeme, aby používateľ dokázal vytvoriť názov vlastného atribútu a aby vedel k nemu priradiť ľubovoľnú hodnotu, ktorá by prislúchala tomuto vytvorenému názvu atribútu. Niekedy môže nastať taká situácia, že názov atribútu sa už nachádza na webovej stránke, preto by sme chceli implementovať aj možnosť priradenia názvu atribútu len kliknutím na názov atribútu. K anotovanej hodnote by sa tento názov priradil a používateľ by už nemusel vypisovať názov atribútu.

Ďalej navrhujeme, aby na detailovej stránke objektu bolo možné anotovať zoznamy rôznych typov objektov. Hlavnému objektu by teda prislúchali zoznamy iných objektov, ako napríklad zoznam hercov pre anotovaný film. Navyše by sme chceli, aby bolo možné pre každý objekt z takéhoto zoznamu priradiť ďalší zoznam objektov alebo atribúty a docieliť stromovú štruktúru.

Chceli by sme, aby takto anotované dáta boli uložené v súbore formátu JSON, pričom ich štruktúra by zodpovedala štruktúre ENode-ov v Exagu. Momentálna implementácia vytvárania XPath-ov v jednotlivých ENode-och funguje tak, že pre vybraný ENode sa najprv zistí XPath rodičovského ENode-u a následne sa k tomuto rodičovskému XPath-u vygeneruje časť XPath-u. Po konkatenácii oboch XPath-ov vznikne XPath pre vybraný ENode. Z toho vyplýva, že je možné priradiť nejakému objektu len také ďalšie objekty, ktoré sa nachádzajú ako potomkovia v HTML elemente tohto objektu. Je teda potrebné implementovať spôsob, ktorým by sa automaticky zistilo pred vytváraním XPath-u pre daný ENode, či sa myšou pred kliknutím nachádzame nad objektom, ktorý anotuje rodičovský ENode, alebo či sa nachádzame mimo objektu, ktorý anotuje rodičovský ENode. Ak sa myšou nachádzame, tak je potrebné vygenerovať taký XPath, ktorý len doplní XPath  rodičovského ENode-u. Ak sa nachádzame myšou mimo objektu, ktorý anotuje rodičovský ENode, tak je potrebné vygenerovať úplne nový XPath bez ohľadu na XPath rodičovského ENode-u.

## Zoznam objektov

Exago dokáže anotovať a extrahovať zoznam komentárov používateľov daného produktu. Každý komentár má atribúty: autor, názov, dátum, popis, hodnotenie produktu a hodnotenie komentára. Štruktúra takto uložených komentárov nie je hierarchická, resp. stromová. Používateľ Exaga nedokáže extrahovať ľubovoľný zoznam objektov s ľubovoľnými atribútmi, avšak extrahovanie komentárov je tomu veľmi podobné.

Navrhujeme vytvoriť štruktúru podobnú komentárom z Exaga, kde si používateľ bude môcť pomenovať atribúty podľa vlastnej potreby, ako je to implementované vo všetkých spomínaných scraper-och. Tým pádom používateľ bude môcť anotovať a extrahovať ľubovoľné objekty s ľubovoľnými atribútmi. V tejto štruktúre by sme chceli vytvárať vzťahy medzi jednotlivými objektmi a atribútmi ako v scraper-i ParseHub a hierarchicky ich ukladať.

Taktiež by sme chceli rozlišovať jednoduchý zoznam hodnôt od zoznamu objektov. V súčasnej verzii Exaga je implementovaná funkcionalita, ktorá jedným kliknutím na jeden text zo zoznamu textov označí celý zoznam týchto súvisiacich textov. Túto funkcionalitu by sme chceli zachovať a vylepšiť ju tým, že takémuto zoznamu súvisiacich textov by sme dokázali priradiť názov.

Zoznam objektov navrhujeme implementovať tak, že používateľ si jedným klikom označí jeden objekt zo zoznamu objektov. Následne sa mu podobne ako v predošlom prípade farebne označia všetky objekty zo zoznamu súvisiacich objektov. Následná anotácia by prebiehala už len v jednom z týchto objektov, pričom zmeny by sa preukázali na všetkých označených objektoch.

## Jeden objekt na viacerých stránkach

Taktiež by sme chceli implementovať funkcionalitu, ktorá umožňuje anotovanie jedného objektu s detailmi zobrazenými na viacerých detailových stránkach. Navrhujeme, aby sa toto anotovanie dalo vykonať podobným spôsobom, ako to robí scraper Data Toolbar, teda pomocou preklikávaním sa medzi stránkami. Navyše by sme používateľovi dali možnosť výberu, či si praje, aby vytvorený klik bol uložený v extrahovaných dátach a k tomuto kliku by prislúchali dáta z webovej stránky, alebo by sa tento klik neuložil. Na úroveň takého kliku by už v extrahovaných dátach boli uložené dáta z webovej stránky, na ktorú klik smeruje.

## Interaktívne anotovanie

Ďalej navrhujeme implementovať možnosť interaktívneho anotovania, ako v scraper-och Dexi.io alebo Data Miner, a tak zjednodušiť prácu anotovania neskúseným používateľom. Cieľom je, aby používateľ po spustení Exaga vedel vykonať anotáciu bez toho, aby študoval návody alebo tutoriály. Túto funkcionalitu by používatelia mohli vypnúť po získaní zručnosti v našom scraper-i.

## Extrakcia

Anotovanie ľubovoľného objektu či zoznamu ľubovoľných objektov nestačí pre univerzálny scraper. Je potrebné, aby sme takto anotované dáta dokázali aj extrahovať. Súčasná verzia Exaga ukladá dáta do formátu JSON, avšak dáta neukladá do plne stromovej štruktúry. Všetky dáta, ktoré sa extrahujú sú na rovnakej úrovni, pričom pre jednotlivý objekt (produkt z eshop-u) je rozdelený na časť atribúty a na časť komentáre.

Extrakciu ľubovoľného objektu s vlastnými názvami atribútov či zoznamu objektov súčasná verzia Exaga nezvládne. Je potrebné vytvoriť novú extrakciu, ktorá takéto objekty dokáže extrahovať, a tým pádom plne využiť potenciál súboru formátu JSON.

## Dynamické načítavanie obsahu

Niektoré webové stránky podporujú dynamické načítavanie obsahu, tzv. nekonečné rolovanie (infinite scrolling), kde pri rolovaní myšou nadol sa na konci stránky zobrazí nový obsah. Preto navrhujeme používateľovi poskytnúť možnosť, aby mohol nakonfigurovať v Exagu, že na anotovanej stránke sa nachádza toto dynamické načítavanie a scraper pri extrakcii dát bude vykonávať toto rolovanie, aby mu bol zobrazený predtým skrytý obsah. Tým pádom budeme vedieť extrahovať viac dát.

## Sťahovanie súborov

Väčšina scraper-ov obsahovala možnosť stiahnutia súboru. Exago ešte sťahovanie ľubovoľného súboru nepodporuje, ale jedným z našich ďalších cieľov je, aby v ňom táto funkcionalita bola implementovaná. Chceli by sme, aby bolo možné stiahnuť čo najviac typov súborov. To by zahrňovalo súbory, ktoré nie sú priamo určené na sťahovanie. Môžu to byť napríklad videá alebo audio súbory z rôznych streamovacích služieb, ako je YouTube či SoundCloud. Súbory by sme vedeli stiahnuť na server, kde by im bol priradený jednoznačný identifikátor. Vo výstupe extrakcie by sme ku každému extrahovanému objektu, ktorý obsahuje aj akciu stiahnutie súboru, priradili tento príslušný identifikátor. Takto by súbory uložené na serveri prislúchali k extrahovaným objektom uloženým vo formáte JSON.

## Vypĺňanie textových polí

V niektorých situáciách sa stáva, že dáta, ktoré chceme extrahovať, sú zobrazené až po autentifikácii používateľa alebo po vyplnení nejakého formulára. Je teda vhodné riešiť aj automatizované vyplnenie textových polí na webových stránkach počas extrahovania. Navrhujeme teda vytvoriť akciu „edit text field“ (uprav textové pole), ktorá anotované textové pole vyplní ľubovoľným textom.

## Anotovanie tabuliek

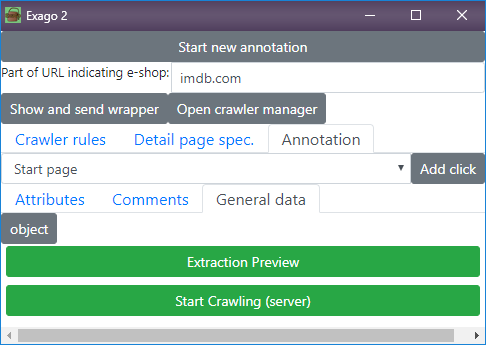
Do nášho scraper-a Exago by sme taktiež chceli implementovať špeciálny modul pre anotovanie ľubovoľných tabuliek, ktoré môžu obsahovať viacero stĺpcov. Navrhujeme, aby anotovanie tabuľky bolo vytvárané takým spôsobom, aby používateľ nemusel ručne písať názvy stĺpcov, ale aby názvy stĺpcov a prislúchajúce hodnoty boli identifikované automaticky. Ak by nastal prípad, že sa názvy týchto stĺpcov identifikujú nesprávne, tak používateľ by mal možnosť anotovať len názvy stĺpcov a konkrétne hodnoty pre každý riadok tabuľky by boli už identifikované automaticky. Používateľ by si mohol vybrať formát, v ktorom by chcel uložiť túto tabuľku, pričom by mal na výber z formátov, ako sú napríklad CSV alebo XLSX.

# Implementácia

## Tlačidlá pre extrakciu a karta „General data“

Pri implementácií nových funkcionalít sme chceli zachovať pôvodné funkcionality, ktorými Exago dokáže anotovať a extrahovať dáta o produktoch z internetových obchodov. Z tohto dôvodu sme vytvorili v Exagu novú kartu s názvom „General data“, kde sme nové funkcionality implementovali. Po spustení Exaga je táto karta prítomná vedľa kariet „Attributes“ a „Comments“.

Po kliknutí na kartu „General data“ sa nám zobrazí tlačidlo „object“, ktorým začneme anotáciu koreňového objektu alebo viacerých koreňových objektov. Exago v tejto konfigurácii môžeme vidieť na nasledujúcom obrázku Obrázok 21.



Obrázok 21 Počiatočný stav karty " General data " a tlačidlá "Extraction preview" a "Start Crawling (server)"

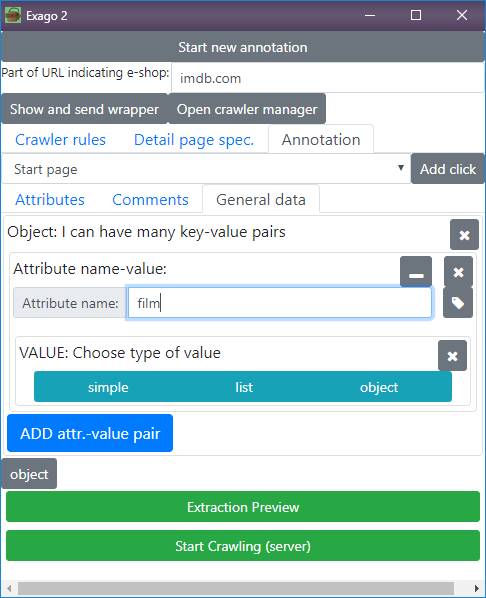
## Anotovanie ľubovoľného objektu

Po kliknutí na tlačidlo „object“ sa nám v Exagu zobrazí ENodeObject. Tento ENode reprezentuje objekt v súbore formátu JSON, ktorý je reprezentovaný kučeravými zátvorkami. Keďže vo formáte JSON sa objekt skladá z viacerých dvojíc kľúč-hodnota, resp. key-value, tak sme do nášho ENodeObject pridali tlačidlo „ADD attr.-value pair“, kde kľúč z takejto dvojice reprezentuje názov atribútu. Toto tlačidlo je prítomné vždy na dolnom konci ENodeObject-u, čo nám umožňuje vytvárať viacero dvojíc kľúč-hodnota pre tento ENodeObject. Kliknutím na toto tlačidlo sa nám v ENodeObject-e zobrazí ENodeAttributeValue.

### Anotovanie názvu atribútu

Prvou z možností ako vyplniť názov atribútu je manuálne napísanie, pri ktorom sa používa textové políčko s názvom „Attribute name“. Bez vyplnenia názvu atribútu nám neprebehne extrakcia pre takýto pár názvu atribútu a jeho hodnoty, preto nemá zmysel nechávať toto políčko nevyplnené.

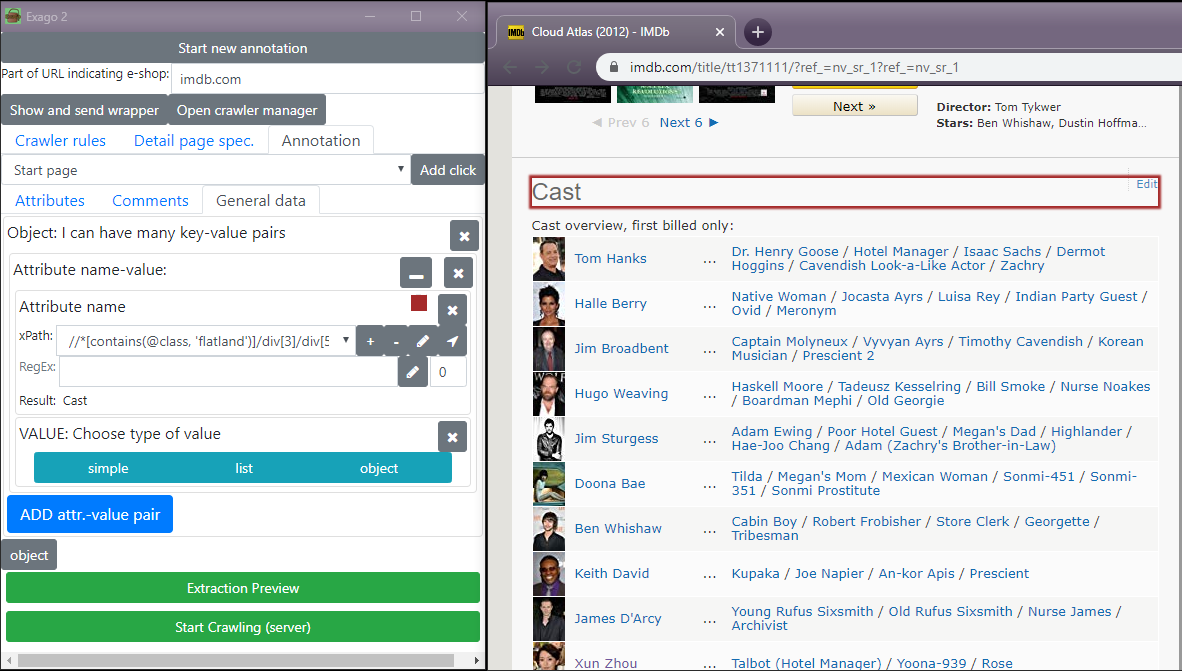
Tento spôsob vypĺňania názvu atribútu je vhodný, ak chceme mať pre objekt extrahovaný na viacerých stránkach rovnaký názov atribútu, napr. pre názov atribútu zvolíme „film“ a hodnota tohto atribútu bude konkrétny názov filmu anotovaný z webovej stránky. Ďalším prípadom použitia tohto spôsobu je, ak sa na webovej stránke nachádza iba hodnota atribútu, ale názov atribútu sa na webovej stránke nenachádza. Tento spôsob si ukážeme na nasledujúcom obrázku Obrázok 22.



Obrázok 22 Manuálne písanie názvu atribútu

Druhou možnosťou ako vyplniť názov atribútu je pomocou anotovania konkrétneho textu na webovej stránke. V tomto prípade musí používateľ najprv kliknúť na tlačidlo so znakom . Následne sa zobrazí ENodeSimple typu „attribute\_name”. V tomto ENode sa nachádza „XPath selector“ a „RegEx selector“, ktorými dokážeme anotovať text z webovej stránky. Po úspešnom anotovaní textu na webovej stránke sa nám v tomto ENode-e zobrazí aj výsledok anotácie, čiže si vieme ľahko overiť jej korektnosť. Ak sa predsa len rozhodneme, že tento spôsob nechceme použiť, tak môžeme zavrieť okno ENodeSimple, následne sa nám opäť zobrazí textové políčko pre manuálne vyplnenie názvu atribútu. Ak sme predtým už vyplnili toto políčko textom, tak bude obsahovať vyplnený text.

Na nasledujúcom obrázku Obrázok 23 si ukážeme tento spôsob vyplnenia názvu atribútu, kde našim cieľom bolo anotovať tabuľku s účinkujúcimi hercami vo filme. Názov atribútu je teda text „Cast“ nachádzajúci sa na webovej stránke.



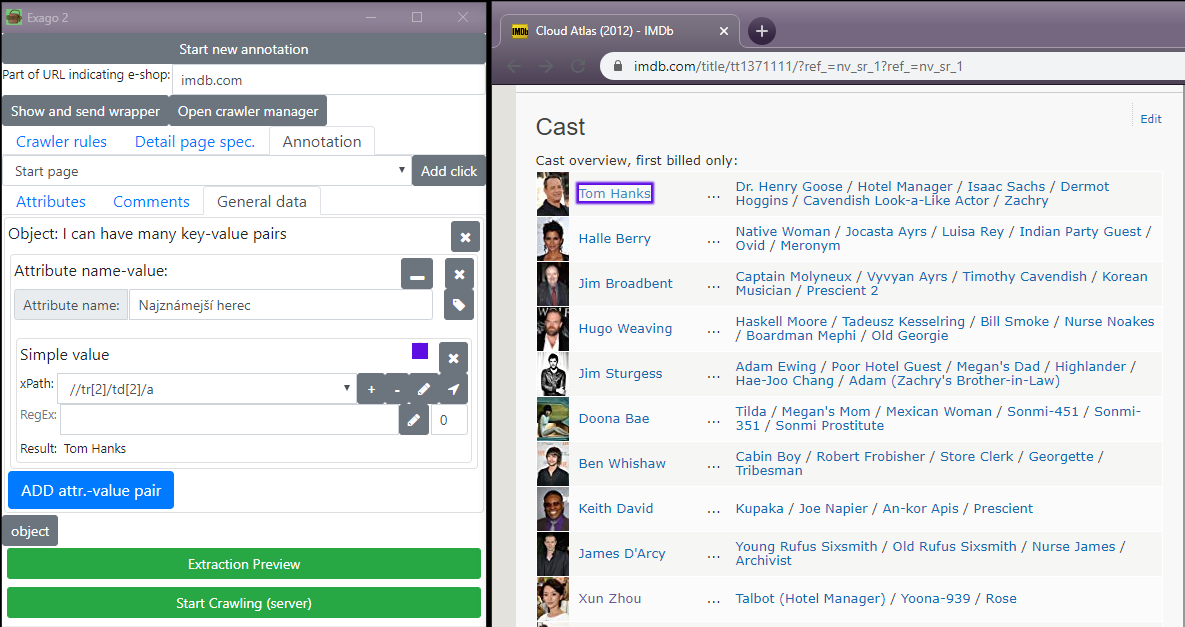
Obrázok 23 Vyplnenie názvu atribútu pomocou klikania myšou

### Anotovanie hodnoty

Ďalšou časťou anotovania dvojice názov atribútu-hodnota je anotovanie hodnoty. Ako hodnotu si môžeme vybrať z troch možností: „simple“, „list“ alebo „object“. V tejto časti opíšeme všetky tri druhy typov hodnôt a spôsoby výberu medzi týmito typmi.

#### Hodnota typu „simple“

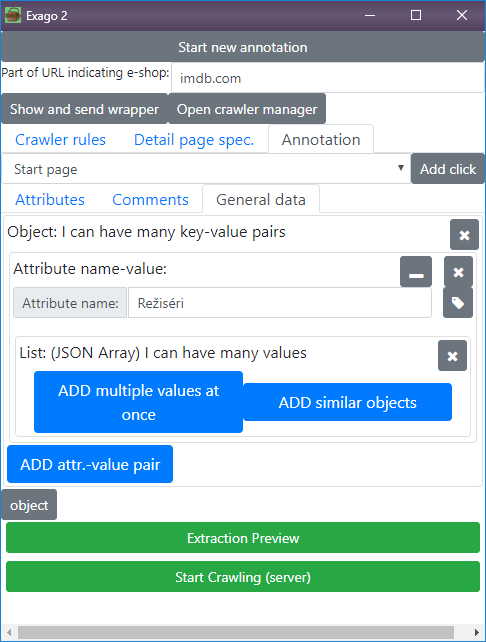
Po kliknutí na tlačidlo „simple“ ENode-u ENodeValue sa zobrazí namiesto tohto ENode-u nový ENodeSimple typu „simple\_value“. Tento typ hodnoty použijeme, ak chceme extrahovať z HTML elementu na webovej stránke jeho text. Po extrahovaní tejto hodnoty sa v súbore formátu JSON tento text zobrazuje na mieste hodnoty. Na nasledujúcom obrázku Obrázok 24 si ukážeme anotovanie najznámejšieho herca filmu, ktorý sa v tabuľke zoznamu hercov nachádza na prvom mieste.



Obrázok 24 Anotovanie hodnoty typu „simple"

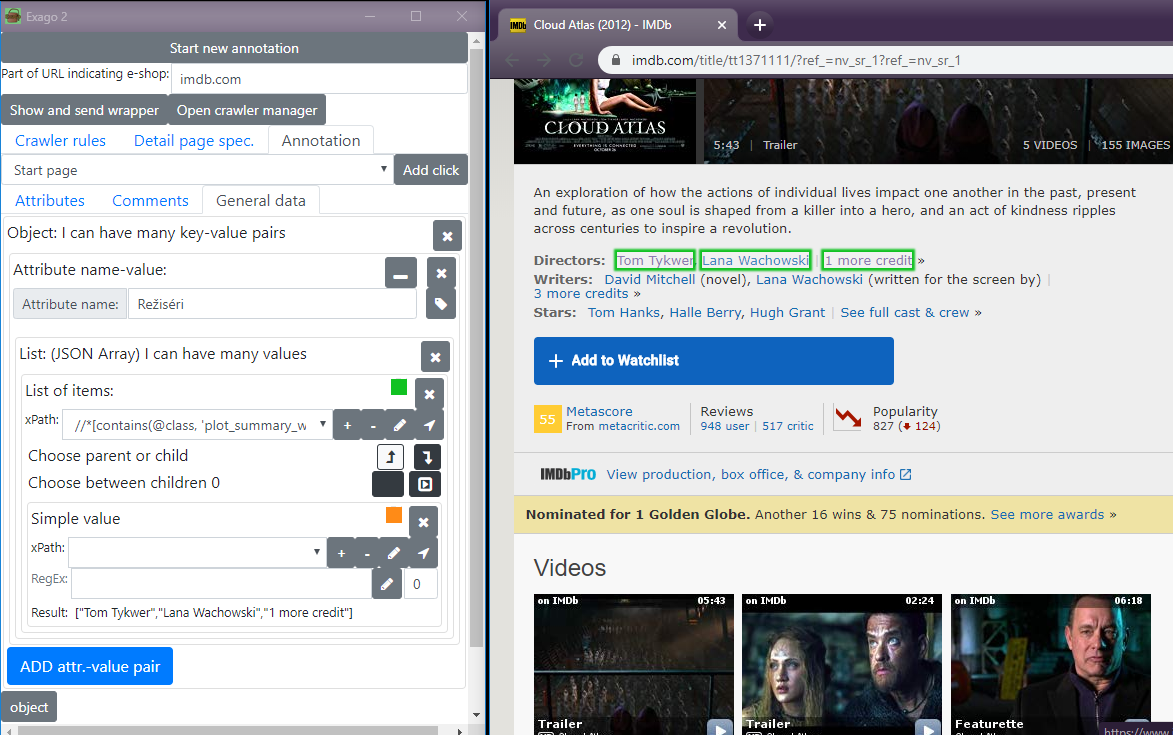
#### Hodnota typu „list“

Po výbere hodnoty typu „list“ sa nám zobrazí ENodeList, ktorý reprezentuje v súbore formátu JSON zoznam. Tento ENode obsahuje tlačidlá „ADD multiple values at once“ a „ADD similar objects“. Tento ENode je zobrazený na nasledujúcom obrázku Obrázok 25.



Obrázok 25 ENodeList s tlačidlami "ADD multiple values at once" a "ADD similar objects"

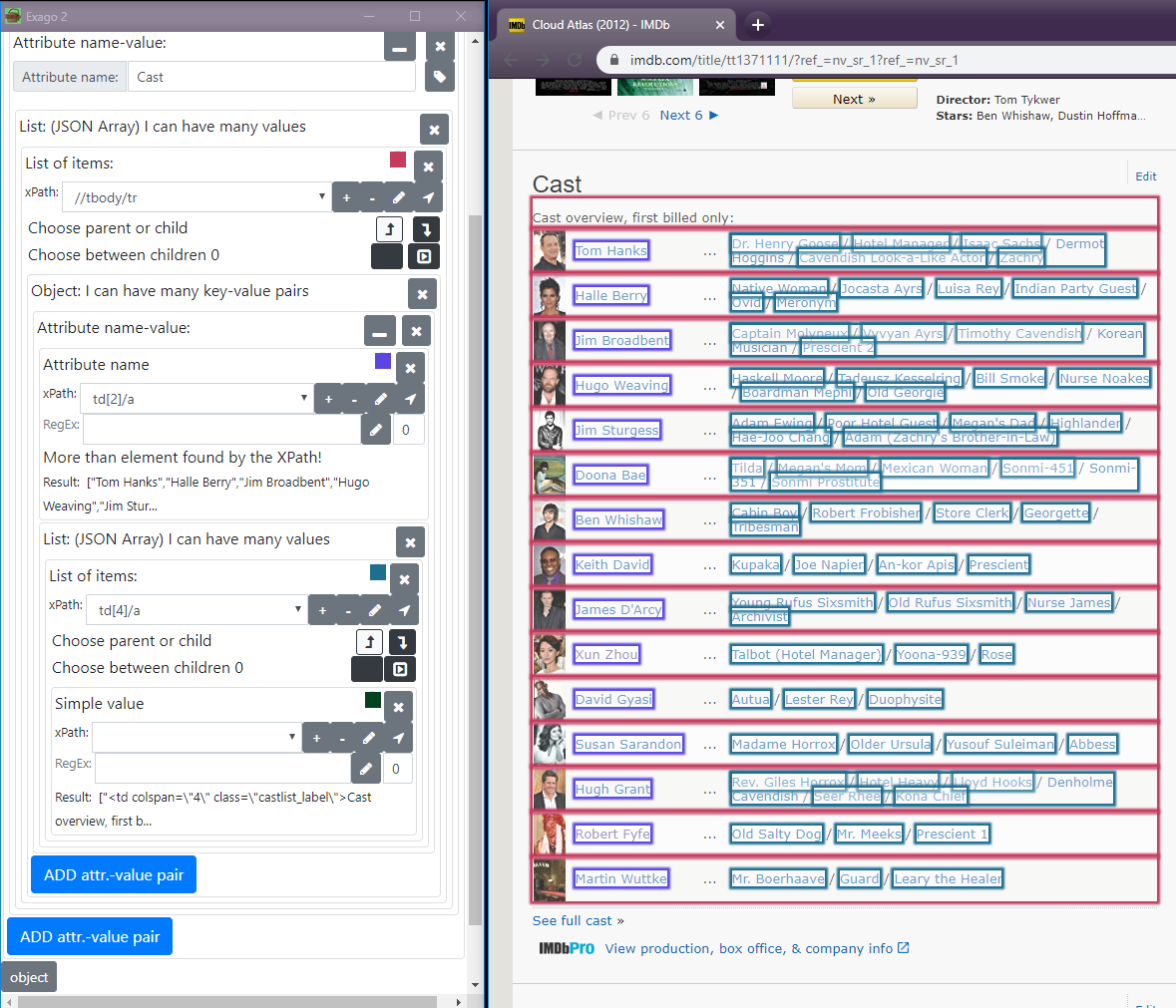
Ak chceme pridať len zoznam textov, ktoré spolu súvisia a nachádzajú sa na webovej stránke pri sebe, tak si používateľ vyberie tlačidlo „ADD multiple values at once“. Následne sa zobrazí ENodeForEach, ktorý sa v Exagu zobrazuje pod názvom „List of items“. Tento ENode zaistí opakovanie jeho vnútorných ENode-ov, pričom sa v našom prípade v ňom nachádza práve ENodeSimple typu „simple\_value“. Pomocou neho vieme teda anotovať zoznam týchto jednoduchých textov kliknutím len na jeden z nich. Následne sa nám označia jednotlivé texty z tohto zoznamu naraz. Na nasledujúcom obrázku si ukážeme anotovanie zoznamu režisérov filmu, pričom názov atribútu sme napísali manuálne slovom „Režiséri“.



Obrázok 26 Anotovanie zoznamu pomocou tlačidla "ADD multiple values at once"

Kliknutím na tlačidlo „ADD similar objects“ sa nám taktiež objaví ENodeForEach, ale v tomto prípade bude obsahovať ENodeObject. Takto vieme vytvoriť zoznam objektov. Používateľ musí najprv označiť objekty na webovej stránke kliknutím na jeden zo zoznamu súvisiacich objektov. ENodeForEach nám označí všetky súvisiace objekty a používateľ môže ďalej anotovať dáta vo vnútri objektov, pričom stačí pracovať iba s jedným objektom. Zmeny sa preukážu aj na ostatných označených objektoch.

Na nasledujúcom obrázku Obrázok 27 budeme anotovať zoznam hercov a priradíme im ich role, ktoré hrali vo filme (v tomto konkrétnom filme každý herec hral viacero rolí). Ako jednotlivý objekt budeme považovať riadok z tabuľky „Cast“. Používateľ klikne na jeden riadok z tejto tabuľky a Exago označí všetky riadky tejto tabuľky, v našom prípade zelenou farbou. Následne pridáme do ENodeObject-u jednu dvojicu názov atribútu-hodnota, pričom ako názov atribútu sme zvolili meno herca označený modrou farbou a hodnotu sme vybrali zoznam rolí označených červenou farbou. Zoznam rolí sme anotovali pomocou tlačidla „ADD multiple values at once“.



Obrázok 27 Anotovanie zoznamu objektov

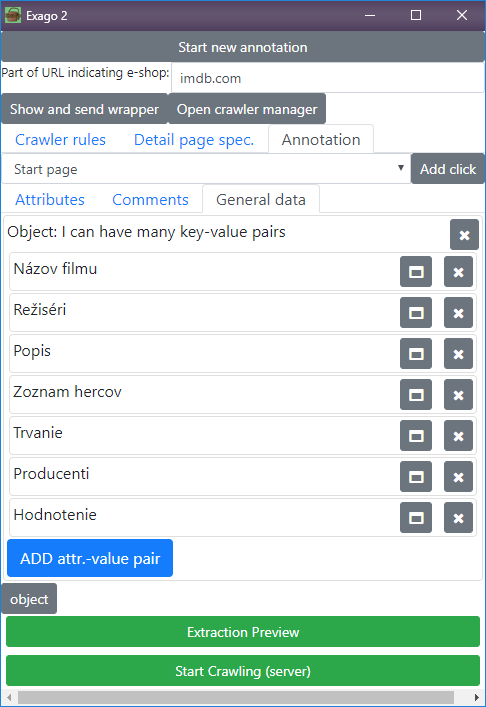
Týmto sme dosiahli stromovú štruktúru, kde máme objekty nachádzajúce sa v hlavnom objekte. Pre hlavný objekt s dvojicou názvu atribútu a hodnoty, máme názov atribútu „Cast“ a hodnotu zoznam objektov. Ďalej sme dosiahli štruktúru zoznam v zozname, pretože každý objekt zo zoznamu objektov má v sebe zoznam rolí hercov. Samozrejme sa môže stať, že namiesto jednoduchého zoznamu rolí, ktorý sme anotovali pomocou tlačidla „ADD multiple values at once“ sa v objekte bude nachádzať ďalší zoznam objektov. Naša implementácia vyrieši aj takúto situáciu použitím tlačidla „ADD similar objects“ a budeme mať teda pre každý objekt zo zoznamu objektov ďalší zoznam objektov.

#### Hodnota typu „object“

Vybraním tlačidla „object“ sa nám otvorí ENodeObject, ktorý bude hodnotou pre danú dvojicu. Tento objekt obsahuje len tlačidlo „ADD attr.-value pair“, s ktorým môžeme pridávať nové dvojice, názvy atribútov a hodnoty. Týmto dosiahneme znovu stromovú štruktúru, kde objekt je vnorený ako hodnota dvojice jeho rodičovského objektu. Samozrejme je možné do tohto objektu vložiť ako hodnotu dvojice opäť ďalší objekt.

### Minimalizácia okna ENodeAttributeValue

Kvôli priehľadnosti anotovania a navigácie v Exagu sme sa rozhodli pridať pre ENodeAttributeValue tlačidlo minimalizácie v hornom pravom rohu. Po kliknutí na toto tlačidlo sa celý obsah tohto ENode-u minimalizuje, pričom sa zobrazí len nadpis tohto minimalizovaného okna s dvomi tlačidlami. Nadpis je nastavený podľa názvu atribútu, ktorý je v ňom obsiahnutý. Tlačidlo minimalizácie sa zmení na tlačidlo maximalizácie, ktorým je možné znovu otvoriť obsah ENode-u. Na nasledujúcom obrázku Obrázok 28 si ukážeme stav, kde sú všetky ENodeAttributeValue daného objektu minimalizované.



Obrázok 28 Ukážka minimalizovania ENodeAttributeValue okien

### Šípky pre anotovanie objektov

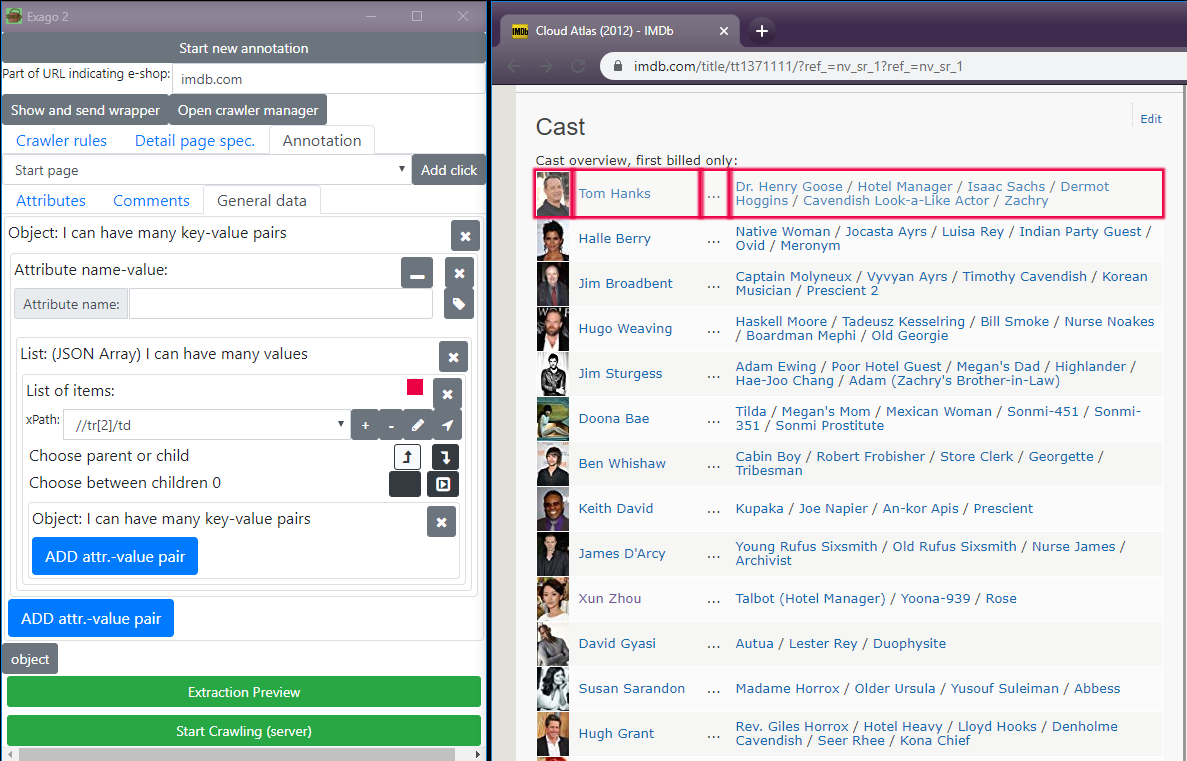
V mnohých prípadoch sa stáva, že nie je možné kliknúť na chcený HTML element na webovej stránke. Buď sa nám podarí označiť nejakého jeho predka alebo niektoré z jeho detí. Používateľ nevie teda nastaviť kurzor myši tak, aby označil chcený HTML element.

Vytvorili sme preto šípky, ktorými je možné sa vynoriť alebo vnoriť do HTML elementov. Používateľ musí najprv kliknúť na element, ktorý chce extrahovať a podľa označenia tohto elementu je potrebné sa buď dostať vynorením k nejakému z predkov označeného elementu pomocou šípky smerujúcej nahor alebo vnorením sa k nejakému potomkovi pomocou šípky smerujúcej nadol.

Každý element má jediného rodiča. Pre vynorenie sa k chcenému predkovi používateľ kliká na šípku smerujúcu nahor, kým nie je označený chcený predok. Na pozadí je to implementované spôsobom, že sa vygenerujú XPath-y k rodičovi aktuálne označeného elementu. Jeden z týchto XPath-ov sa nastaví do x-path-selector-u daného ENode-u a prefarbí sa označenie na webovej stránke. Používateľ tak môže skontrolovať správnosť anotácie. Ak sa nevynoril dostatočne, tak môže tento postup opakovať.

Pre prípad vnorenia sa k potomkom označeného elementu je situácia náročnejšia. Dôvodom je to, že HTML element môže obsahovať viacero detí. Je preto potrebné vybrať si, ktoré z jeho detí chceme označiť. Používateľ teda šípkou smerujúcou nadol sa presunie do úrovne detí pôvodne označeného elementu a následne šípkami smerujúcimi doprava a doľava si vyberá to správne dieťa. Implementácia je robená takým spôsobom, že sa na pozadí vygenerujú XPath-y pre každé dieťa pôvodne označeného elementu. V x-path-selector-e sa menia skupiny XPath-ov pre každé vybrané dieťa a tieto skupiny XPath-ov sa menia pomocou šípok smerujúcich doľava a doprava. Na webovej stránke sa podľa aktuálne vybraného XPath-u automaticky prefarbujú elementy.

Táto funkcionalita je implementovaná iba pre ENodeForEach, ktorý je v Exagu nazývaný „List of items“. Pod jeho x-path-selector-om sa nachádzajú dva informačné texty pre prácu so šípkami. Vedľa textu „Choose between children“ je číslo, ktoré znázorňuje počet detí, medzi ktorými sa po kliknutí na šípku smerujúcu nadol pohybujeme šípkami smerujúcimi doľava a doprava. Ak používateľ narazí na prvé dieťa zo zoznamu a klikne na šípku smerujúcu doľava, tak sa mu zobrazí okno, že je na začiatku zoznamu a ďalšie dieťa sa už tým smerom nevyskytuje. To isté platí aj pri poslednom dieťati zo zoznamu, keď používateľ klikne na šípku smerujúcu doprava. Na nasledujúcom obrázku Obrázok 29 chcel používateľ označiť ako objekty každý riadok tabuľky zo zoznamu hercov. Ukážeme si, že to nebolo možné a po kliknutí na herca Tom Hanks sa označili ako riadky tabuľky iba časti jedného riadka tabuľky. Používateľ by sa teda chcel v tomto prípade vynoriť o niekoľko úrovni vyššie, až kým by červenou farbou nemal označené všetky riadky tabuľky.



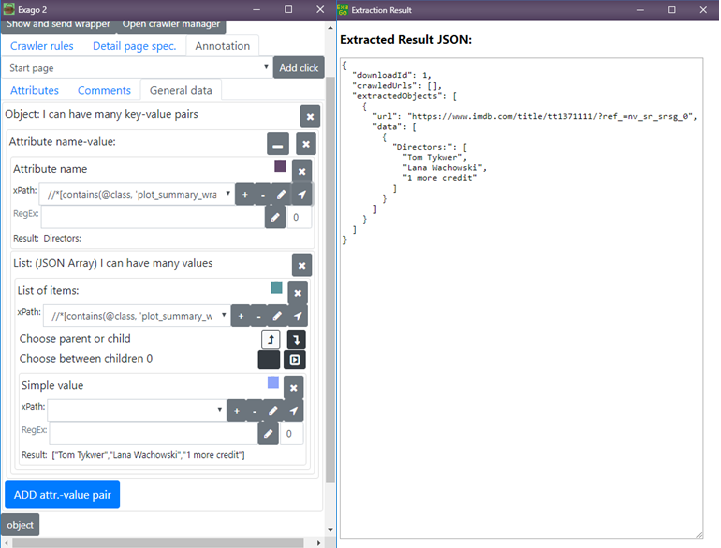
Obrázok 29 Prípad, keď používateľ nedokáže jediným kliknutím označiť chcené elementy

## Extrakcia ľubovoľných objektov

Implementovaním novej anotácie, ktorá dokáže vytvárať objekty vnorené v iných objektoch a zoznamy vnorené v iných zoznamoch je predošlá implementácia extrakcie nedostatočná. Použitím nových ENode-ov a ich usporiadania do rôznych stromových štruktúr sa predošlá extrakcia nevie s týmito funkcionalitami vysporiadať. Bolo teda potrebné vytvoriť novú extrakciu, ktorá by tieto ENode-y zvládla rekurzívne extrahovať. Pôvodnú extrakciu a náhľad extrakcie sme ponechali, pretože tá je špecificky implementovaná pre eshop-y.

### Tlačidlo „Extraction preview“

Tlačidlom „Extraction preview“ si dokážeme prezrieť, ako budú vyzerať dáta po extrahovaní z webovej stránky, na ktorej sa nachádzame. Po kliknutí na toto tlačidlo sa nám otvorí nové okno s názvom „Extraction Result“, kde sa bude nachádzať súbor formátu JSON s extrahovanými dátami. Toto tlačidlo je vhodné použiť až po anotácii webovej stránky, pretože inak by sa v náhľade zobrazili iba základné dáta extrakcie, čo sú identifikátor sťahovania, crawlované webové stránky a koreňový extrahovaný object s vyplnenou URL adresou stránky, na ktorej sa nachádzame. Na nasledujúcom obrázku Obrázok 30 sa nachádza okno „Extraction Result“, pričom sme na webovej stránke anotovali jeden objekt. Tento objekt obsahuje jedinú dvojicu názov atribútu-hodnota, pričom ako názov atribútu sme označili „Directors“ a hodnota tejto dvojice je zoznam režisérov z webovej stránky. Extrahovaný objekt sa nachádza ako hodnota atribútu „data“ koreňového extrahovaného objektu.



Obrázok 30 Okno „Extraction Result" s náhľadom na extrahované dáta

Na pozadí tohto tlačidla sa deje lokálna extrakcia dát z webovej stránky, na ktorej sa nachádzame. To sa neskôr použije ako základ extrakcie pre celý internetový portál, ktorá bude spustená tlačidlom „Start Crawling (server)“.

Lokálna extrakcia webovej stránky, na ktorej sa nachádzame, je implementovaná tak, že na začiatku extrakcie sa vyrobí jeden koreňový objekt s dvojicou atribútov „URL“ a „data“. Tieto atribúty budú neskôr figurovať vo výslednom JSON súbore ako dvojica názov atribútu a hodnota, pričom hodnota pre názov atribútu „URL“ je webová stránka, na ktorej sa nachádzame. V atribúte „data,“ ktorého hodnota je typu pole, sa budú nachádzať extrahované objekty.

Jednotlivé extrahované objekty v atribúte „data“ sú tie objekty, ktoré boli vyrobené v Exagu pomocou tlačidla „object“. Tieto objekty, ktoré sú typu ENodeObject, sa extrahujú rekurzívne tak, že sa pre každý objekt vnárame do ich detí. Tieto deti sú taktiež ENode-y, pre ktoré sa rovnako spúšťa rekurzívna extrakcia. Všetky ENode-y vytvárajú špecifické štruktúry pre výsledný JSON súbor.

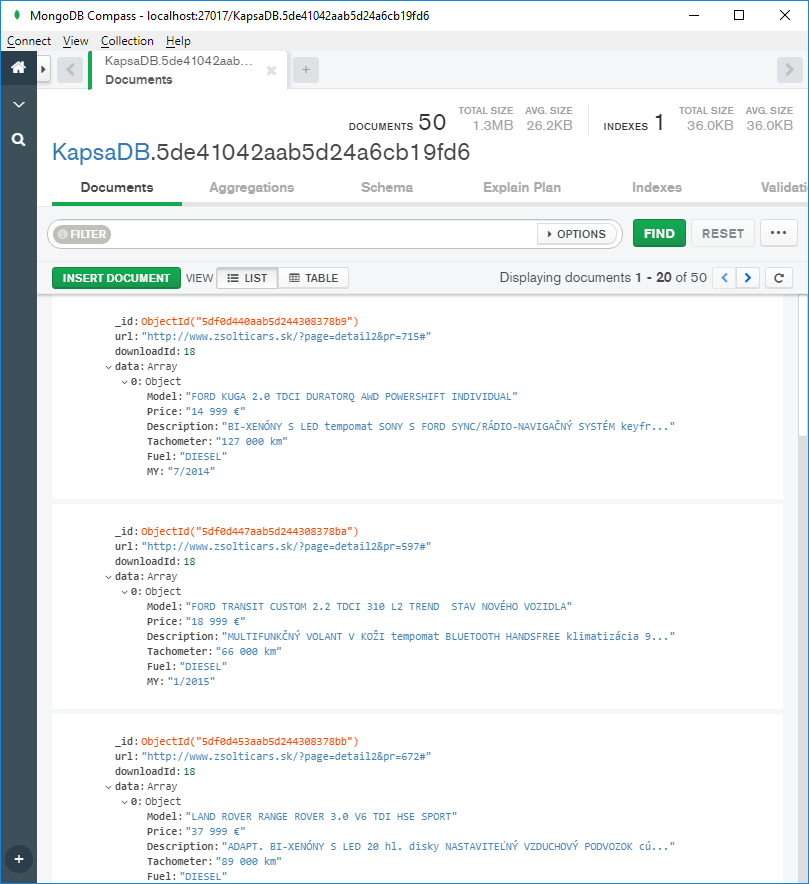
ENodeObject vytvorí kučeravé zátvorky a obsahuje niekoľko ENode-ov typu ENodeAttributeValue. ENodeAttributeValue dodá do extrakcie názvy atribútov, ak je napísaný názov atribútu manuálne. Ak je názov atribútu vytvorený pomocou anotácie, tak ENodeAttributeValue spustí rekurzívnu extrakciu pre ENodeSimple typu „simple\_attributeName“. Tento ENode už nemá deti a dodá do výsledku extrakcie názov atribútu z webovej stránky. Následne sa spustí rekurzívna extrakcia pre ENode, ktorý je hodnotou ENode-u ENodeAttributeValue. Prípad, keď je tento ENode reprezentujúci hodnotu typu ENodeObject, sme už popísali.

Ak je hodnotou ENode ENodeSimple typu „simple\_value“, tak sa z webovej stránky extrahuje hodnota, ktorú anotuje a rekurzia pre rodičovský ENodeAttributeValue končí. Pre ENodeObject, ktorý je rodičom tohto ENode-u sa znovu spustí rekurzia pre ďalší ENodeAttributeValue.

Môže sa stať, že hodnotou ENodeAttributeValue bude ENodeList. V JSON súbore sa tak vytvoria hranaté zátvorky reprezentujúce pole. Ak je dieťaťom ENodeList-u ENodeForEach, ktorý má dieťa ENodeSimple typu „simple\_value“, tak v súbore JSON budú medzi vytvorenými hranatými zátvorkami jednoduché texty v úvodzovkách. Tieto jednoduché texty sú extrahované z webovej stránky. Ak je dieťaťom ENodeList-u ENodeForEach, ktorý má dieťa ENodeObject, tak sa pre každý ENodeObject spustí rekurzívna extrakcia, ako sme popísali vyššie.

### Tlačidlo „Start crawling (server)“

Týmto tlačidlom sa spustí extrakcia celého portálu, pričom Exago bude zo servera dostávať URL stránky, ktoré má navštíviť a znova spúšťať na nich lokálnu extrakciu popísanú v predošlej časti. Výsledky týchto lokálnych extrakcií sa posielajú do databázy MongoDB, ktorá sa nachádza na serveri. Na nasledujúcom obrázku Obrázok 31 môžeme vidieť extrahované dáta nachádzajúce sa v MongoDB. Tieto dáta o ojazdených autách boli extrahované z autobazáru [www.zsolticars.sk](http://www.zsolticars.sk), pričom boli extrahované dáta 50-tich áut.



Obrázok 31 Výsledok extrakcie na server do databázy MongoDB z autobazáru [www.zsolticars.sk](http://www.zsolticars.sk)