



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Факультет Компьютерных Наук
Департамент Программной Инженерии
Курсовая работа

Клиент-Серверное Приложение для Управления Скидками в Розничных Сетях

Выполнили студенты гр.БПИ151
Куприянов Кирилл
Суровцев Максим
Научный руководитель:
Профессор ДПИ Александров Дмитрий Владимирович

2018

Агрегаторы скидок - место, где собрана вся информация об акционных товарах и выгодных предложениях.



1. Crawler — Программный модуль, работающий в фоне и производящий сбор данных с сайтов указанных магазинов, с последующей отправкой их на сервер в формате JSON
2. REST API — Стил ь архитектуры программного обеспечения для построения веб-сервисов с использованием протокола HTTP
3. ORM — технология программирования, которая связывает базы данных с концепциями объектно-ориентированных языков программирования, создавая «виртуальную объектную базу данных»
4. Activity — компонент приложения, с которым пользователи могут взаимодействовать для выполнения каких-либо действий

Обоснование актуальности работы

Популярность скидок в интернете



Статистика запроса “скидки” в Google



Статистика запроса “Едадил” в Google



1. Неудобство пользования бумажными каталогами
2. Отсутствие приложений с функционалом создания нескольких списков покупок
3. Отсутствие приложений с функционалом добавления в список “пользовательских позиций”
4. Рост количества страниц со скидками на отдельных сайтах магазинов
5. Разный дизайн и архитектура сайтов магазинов

Цель работы — реализовать модульные и масштабируемые web и Android приложения для работы со множеством скинсов покупок, просмотра и управления акциями в розничных сетях

Задачи работы

1. Изучить существующие аналоги данного приложения
2. Выбрать технологии для реализации
3. Продумать модульность архитектуры
4. Реализовать программу
5. Разработать техническую документацию

Едадил — акции в магазинах (бета)



Удобство использования

- Неограниченное количество списков покупок
- В список можно даже добавить товар, которого нет в магазине
- Использование одного аккаунта несколькими людьми

Архитектура

- Модульность
- Масштабируемость
- Поддерживаемость

- Собирает данные с сайтов
- Селекторы: xpath
- Разбит на модули
 - spiders (1 паук на магазин)
 - selectors (1 селектор на магазин)
 - text_processors
 - pipelines



CATEGORIES = '//a[@class="catalog-categories__link"]/@href'

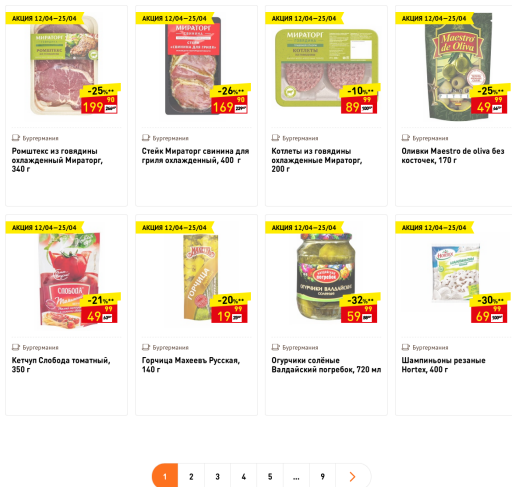
```
<li class="xf-catalog-categories__item">
  <a class="xf-catalog-categories__link" href="/catalog/moloko-syr-yaytsa">
    <span class="xf-catalog-categories__text">Молоко, сыр, яйца</span>
  </a>
</li>
<li class="xf-catalog-categories__item">
  <a class="xf-catalog-categories__link" href="/catalog/ovoschi-frukty-griby">
    <span class="xf-catalog-categories__text">Овощи, фрукты, грибы</span>
  </a>
</li>
<li class="xf-catalog-categories__item">
  <a class="xf-catalog-categories__link" href="/catalog/myaso-ptitsa-delikatesy">
    <span class="xf-catalog-categories__text">Мясо, птица, деликатесы</span>
  </a>
</li>
<li class="xf-catalog-categories__item">
  <a class="xf-catalog-categories__link" href="/catalog/zamorojennye-produkty">
    <span class="xf-catalog-categories__text">Замороженные продукты</span>
  </a>
</li>
```

Алгоритм работы программы

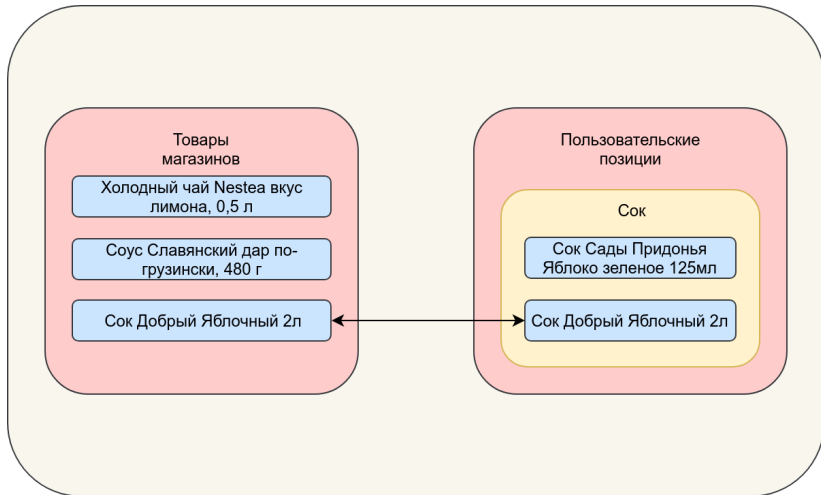
Crawler: пауки (spiders)

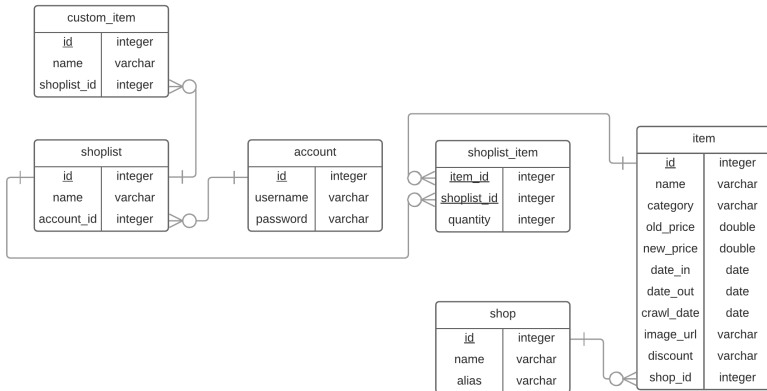


1. Открывает страницу сайта (загружает исходный код)
2. Ищет корневой элемент
3. Создаётся Item
4. Последовательно заполняет поля
5. Отправляет Item в модуль pipelines
6. Переходит на следующую страницу



```
class DataBaseWriterPipeLine(object):  
  
    def process_item(self, item, spider):  
        requests.post(const['ADD_ITEM_API'],  
                      data=json.dumps(dict(item)),  
                      headers=const['REQUEST_HEADERS'])  
        return item
```







/api/shops/1?category=Напитки&page=1

```
{
  "count": 9,
  "numPages": 1,
  "rows": [
    {
      "id": 142138,
      "name": "Холодный чай Nestea вкус лимона, 0,5 л",
      "category": "Напитки",
      "oldPrice": 66.2,
      "newPrice": 44.99,
      "dateIn": "2018-04-09",
      "dateOut": "2018-04-15",
      "crawlDate": "2018-04-11",
      "condition": "-",
      "image": null,
      "imageUrl": "https://dixy.ru/upload/iblock/ea9/2000003636.jpg",
      "discount": "-32",
      "shop": {
        "id": 1,
        "alias": "dixy",
        "name": "Дикси"
      }
    }
  ]
}
```

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJpZCI6MiwiZXNlcm5hbWUiOiJyb290IiwiaWF0IjoxNTIzNjgyNTMyfQ.
OvECDXoD-IU0AUrP99gGxyYjMxsoIFiK4EBsDVQHAfo

Header

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

Payload

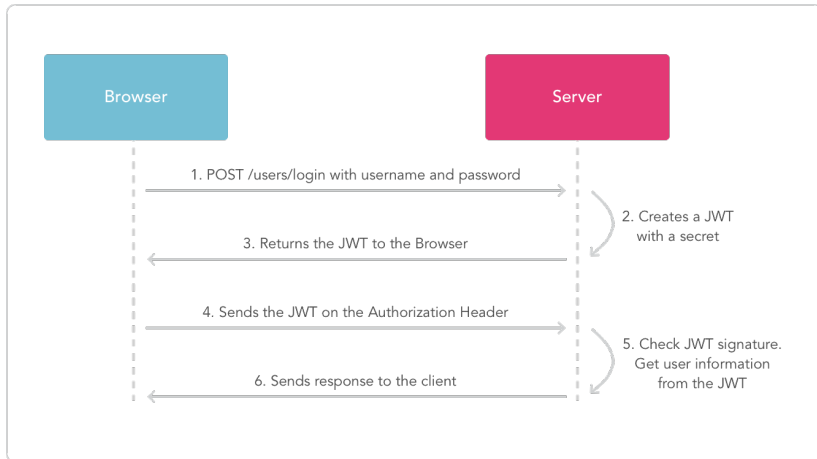
```
{  
  "id": 2,  
  "username": "root",  
  "iat": 1523682532  
}
```

Signature

```
hash(  
  base64(header) + "." +  
  base64(payload),  
  secret)
```


Алгоритм работы программы

JWT







[4]



Демонстрация



- Больше магазинов
- Сотрудничество с магазинами
- Отображение магазинов на карте
- Подписка на уведомления о новых акциях
- Отправка списков покупок между пользователями
- Клиенты на других платформах (IOS)

-  *Android Developers references and guides.* 1 февр. 2018. URL: <https://developer.android.com/index.html>.
-  *Google Trends - latest trends, data and visualizations.* 10 апр. 2018. URL: <http://trends.google.com/>.
-  *Google's material design color guideline.* 6 февр. 2015. URL: <https://material.io/guidelines/style/color.html>.
-  *JSON Web Tokens.* 17 марта 2018. URL: <https://jwt.io>.
-  *Manual | Sequelize | The node.js ORM for PostgreSQL, MySQL, SQLite and MSSQL.* 5 марта 2018. URL: <http://docs.sequelizejs.com>.
-  *Scrapy 1.5 documentation.* 11 янв. 2018. URL: <http://scrapy.readthedocs.io/en/latest/index.html>.

Факультет Компьютерных Наук
Департамент Программной Инженерии
Курсовая работа

Выполнили студенты гр.БПИ151
Куприянов Кирилл
Суровцев Максим
Научный руководитель:
Профессор ДПИ Александров Дмитрий Владимирович

2018

www.4pda.ru/forum/index.php?showtopic=897451

www.gcsales.ru