Фамилия Имя Отчество автора

**НАЗВАНИЕ ТЕМЫ**

Междисциплинарная курсовая работа

по направлению подготовки

09.03.01. Информатика и вычислительная техника

студента образовательной программы   
«Информатика и вычислительная техника»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Студент** | |  | **Руководитель** | |
| / |  |  | / |  |
| *подпись* | *И.О. Фамилия* |  | *подпись* | *должность, звание, И.О. Фамилия* |
|  | |  | **Консультант** | |
|  |  |  | / |  |
|  |  |  | *подпись* | *должность, звание, И.О. Фамилия* |

Задание на МКР

График выполнения

Аннотация

Оглавление

Введение

В своей работе я ставлю цель перед собой - разработать ПО для параметрического поиска статей на сайте <https://arxiv.org> с возможностью дальнейшего скачивания.

arXiv e-Print Archive основан в 1991 университетом Cornell University и имеет более миллиона научных на тему математики и компьютерных наук.

The Cornell University e-print arXiv, hosted at [arXiv.org](http://arxiv.org/), is a document submission and retrieval system that is heavily used by the physics, mathematics and computer science communities. It has become the primary means of communicating cutting-edge manuscripts on current and ongoing research. The open-access arXiv e-print repository is available worldwide, and presents no entry barriers to readers, thus facilitating scholarly communication. Manuscripts are often submitted to the arXiv before they are published by more traditional means. In some cases they may never be submitted or published elsewhere, and in others, arXiv-hosted manuscripts are used as the submission channel to traditional publishers such as the American Physical Society, and newer forms of publication such as the Journal for High Energy Physics and overlay journals.

Мое программное решение предлагает скачивать статьи большими объемами сразу, не прибегая к веб-версии сайта, что является большой экономией времени.

При этом я ставил перед собой цель - сделать программу удобной в использовании.

Разработка модуля отправки запроса на сайт и получения ответа

У сайта существует программный интерфейс (API), который предоставляет доступ ко всем данным сайта и позволяет выполнять параметрический поиск. Все вызовы выполняются через HTTP GET или POST запросы.

Так как программный интерфейс построен на HTTP, то для отправки запроса на сайт нужно сформировать ссылку, в которой будут заключены нужные нам параметры, по которым будет выполняться поиск на сервере. Сервер вернет нам ответ в виде XML файла и будет содержать результат поиска статей, а также информацию о них. Часть этой информации будет отображаться в Графическом интерфейсе пользователя.

Прочитав и ознакомившись с программным интерфейсом сайта, я выделил параметры, по которым возможен поиск:

|  |  |
| --- | --- |
| **Название** | **Сокращение** |
| all | all |
| Author | au |
| Title | ti |
| Abstract | abs |
| Comment | co |
| Id | id |
| Journal Reference | jr |
| Report number | rn |
| Subject category | cat |

По этим же полям будет граббер будет формировать запрос на сервер.

При этом API предоставляет возможность создания логических условий на то или иное поле. Эта функция также должна быть реализована в программе.

Стоит учесть то, что в ответе от сервера не могут содержаться абсолютно все результаты поиска из-за большого количества статей. В противном случае запросы бы обрабатывались долгое время и каждый ответ имел бы большой вес. Если результатов поиска много, то сервер отправит только часть. Однако мы сами можем определить количество.

Для того, чтобы иметь доступ ко всем статьям из граббера мы будем получать статьи порциями. Я определил максимальное количество в 100 статей за запрос. Так запросы будут обрабатываться программой быстро и не будет сильной нагрузки на графический интерфейс.

Получив ответ от сервера, его необходимо обработать и привести в удобное для программы представление. Для этих целей я написал парсер. Он получает на вход ответ от сервера в виде XML файла и создает экземпляр классов Article для каждой статьи.

Класс Article имеет такую же модель, как и XML файл. Все информация о статьях теперь доступна программе, и она может отобразить эту информацию на экране.

