

задач, будь то резервное копирование, чтение электронной почты и т.д. Язык программирования Python практически ничем не ограничен, поэтому также может использоваться в крупных проектах. К примеру, python интенсивно применяется IT-гигантами, такими как, например, Google и Yandex.

Заключение. На основании сделанных исследований, выявлено, что сервис, который планируется к разработке, будет пользоваться спросом, так как разрабатываемая система предоставляет больше услуг, чем аналоги, поэтому дальнейшая разработка является актуальной. За счет дополнительных функций, что упростят поиск одежды пользователю, таких как калькулятор размеров и система профилей, что хранит мерки пользователя и использует их для рекомендации товара будущим покупателям.

Список использованных источников:

1. Блог-хостинг. [Электронный ресурс]. – Режим доступа: <https://www.insales.ru/blogs/university/tovarnyy-agregator> (Дата обращения 21.02.2020).
2. Блог-хостинг. [Электронный ресурс]. – Режим доступа: <https://spark.ru/startup/z-price/blog/52492/prajs-agregatori-vs-sajti-internet-magazinov-kakoj-nailuchshij-istochnik-dlya-sbora-dannih-o-tovarah> (Дата обращения 21.02.2020).
3. Агрегатор. [Электронный ресурс]. – Режим доступа: <https://market.yandex.ru> (Дата обращения 21.02.2020).
4. Агрегатор. [Электронный ресурс]. – Режим доступа: <https://www.e-katalog.ru> (Дата обращения 21.02.2020).

РАЗРАБОТКА ВЕБ-РЕСУРСА «RANDMOVIE»

Ю.В. Рябов^а, студент группы 17В60,

научный руководитель: Захарова А.А., д.т.н.

Юргинский технологический институт (филиал) Национального исследовательского

Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

^аE-mail: yura-ryabov@mail.ru

Аннотация: Статья посвящена разработке веб - ресурса RandMovie, тому зачем он нужен, посредством каких технологий он будет реализован и какой алгоритм был выбран в системе рекомендаций, а также тому, как пользователь выбирает фильм для просмотра.

Ключевые слова: рекомендации, веб - ресурс, алгоритм, веб - программирование, html5, css3, JavaScript, PHP, SQL.

С каждым годом появляется все больше и больше картин кинематографа, как различных жанров, так и различного качества, так же картины одного жанра, в различных странах могут быть совершенно разные, так как каждой стране присущ свой собственный стиль.

Актуальность веб - ресурса RandMovie заключается в необходимости разработки ресурса, который бы позволил пользователю выбрать фильм или же сериал для просмотра и сделать это как можно более удобно, так как уже существующие аналоги страдают от достаточно большого количества проблем в следствии чего подобные ресурсы становятся не значимыми. Самой же распространённой проблемой подобных ресурсов является неудобность его использования, отсутствие какого адекватного дизайна или же наоборот интерфейс перегружен, а также не всегда предоставляется возможность быстро получить доступ к подобному ресурсу. Даже несмотря на то, что у аналогов может быть огромная база данных с фильмами, в результате она может вообще никак не обрабатываться и просто не использоваться в полную силу.

Веб - ресурс RandMovie можно назвать инструментом, посредством которого любой пользователь может найти фильм по своему вкусу, посредством различной обработки создаваемой собственной базы данных. Получить полную статистику о пользователях ресурса, а так получить автоматически рекомендации, от сторонних пользователей, которые, ближе всего по духу к нему или же вступить в сообщество, которое интересуется таким жанром или жанрами фильмов, которые интересуют и пользователя.

Что бы реализовать подобный ресурс, необходимо прибегнуть к веб - программированию, что в свою очередь является одним из разделов программирования, которое ориентированно на разработку веб - ресурсов и веб - приложений. Различные языки веб - программирования в основном направлены на работу с разнообразными веб - технологиями, например html5 или NodeJs. Языки веб - программирования можно поделить на такие группы как:

Клиентские языки - программирования, которые в основном обрабатываются на стороне пользователя в браузере, но может быть и иначе. Это создает серьезную проблему клиентских языков, так как результат выполнения кода полностью зависит от браузера пользователя, от его типа, какой движок он использует или даже от его версии. То есть, весьма вероятно, может произойти такое, что в разных браузерах или в разных версиях браузера, скрипт может вести себя по-разному или же не работать вовсе.

Серверные языки - из названия следует, что данные функции, методы и т.д., которые обрабатываются со стороны сервера. То есть, когда пользователь создает запрос на странице веб - ресурса или веб - приложения посредством тех же самых серверных языков или же иных, то вызванная страница сначала обрабатывается на сервере и только после обработки возвращается к посетителю в виде файла. Например, таким языком является PHP.

Для разработки ресурса RandMovie необходимо будет прибегнуть к обоим пунктам. Из чего следует, что frond - end часть ресурса следует разрабатывать на таких языках как: html5, CSS3 и JavaScript. А back - end на таких языках как PHP и SQL.

Для каждого зрителя, процесс принятия решения представляет собой интеллектуальную деятельность, принимая решение, зритель мыслит, иначе говоря, решает какую – либо задачу, под воздействием на зрителя различных факторов. Данные факторы выражаются в том, что окружает зрителя, это могут быть друзья или знакомые, которые опираясь на свое личное оценочное суждение, рассказывая о каком – либо фильме повлияют на выбор зрителя по так или иначе. Также повлиять на выбор фильма может его реклама, под воздействием которой зритель так или иначе уже оценивает рекламируемый ему фильм, проще говоря, реклама заставляет зрителя принимать какое – либо решение относительно себя, понимает-ся, чем ближе реклама к зрителю по вкусу, тем более действенной она будет.

Повлиять на выбор зрителя также может и оценка фильма на каком – либо агрегаторе, опираясь на оценку других не знакомых зрителю пользователей агрегатора, зритель может сделать соответствующий выбор. Подобные агрегаторы могут повлиять на выбор фильма зрителем не только посредством оценки сторонних пользователей, но и рецензиями, так как обычно они собирают рецензии со сторонних ресурсов либо сами критики оставляют рецензии на подобных агрегаторах, а также в некоторых случаях сами пользователи могут написать рецензию о каком - либо фильма. Опираясь на данные рецензии, зритель может сделать выбор в сторону того или иного фильма.

Но самым сильным фактором влияния на выбор зрителем того или иного фильма для просмотра, можно считать видео обзоры, на различных видео хостингах, так как в современном мире, доверие как к рекламе, так и к рецензиям стремительно падает, а к обзорам растет.

Условно, описанный выше процесс можно на разделить на отдельные этапы:

Мониторинг - этап во время которого происходит сбор информации о фильме из различных источников, это может быть реклама, видео обзоры или же рецензии кинокритиков, а также различные ресурсы, посвящённые кинематографу.

Анализ - этап во время которого происходит анализ полученной информации, например о фильме, безусловно анализ происходит на основе парадигм вкусов зрителя, который выбирает фильм. Также в данный пункт можно отнести и видео обзоры, так как в своей основе они уже сами по себе представляют некий анализ, но который не основывается на предпочтениях зрителя.

Выбор фильма - этап во время которого зритель выбирает фильм для просмотра на основе мониторинга и анализа.

Оценка - этап оценивания фильма зрителем после его просмотра, исходя из того насколько зрителю понравился фильм, он может его оценить каким либо образом, но также на оценку могут повлиять реклама или рецензии с видео обзорами в той или иной степени.

Для того, чтобы ресурс мог учитывать пожелания пользователей, ресурс должен обладать такими функциями как, учет пользователей, учет фильма, учет пользовательских оценок, а также, что более важно иметь систему рекомендаций задача которой проинформировать пользователя о фильме, который может быть ему интересен в данный момент времени. Система рекомендаций само собой имеет основной набор характеристик, таких как предмет рекомендации, это то, что рекомендуется, в случае разрабатываемого ресурса, это фильмы. Цель рекомендации, то для каких целей рекомендуется. Контекст рекомендации, источник рекомендации, степень персонализации, прозрачность, формат рекомендации. А также алгоритмы, посредством которых происходит подбор рекомендаций. Такие алгоритмы как:

Неперсонализированные рекомендации, самый простой из алгоритмов, в котором интерес пользователя определяется средним рейтингом фильма. Такой алгоритм как правило реализуют

на большинстве сервисов, когда пользователь не нуждается в авторизации на сервисе, и работает данный алгоритм по типу «всем нравится, значит может понравится и вам».

Content based рекомендации являются персональными рекомендациями, алгоритм которых предполагает использование информации о самом пользователе, на разрабатываемом ресурсе RandMovie подобной информацией являлась бы жанры фильма которые предпочитает пользователь. То есть в рамках данного алгоритма описание фильма сопоставлялось бы с интересами пользователя, полученных, например их выставленных ранее оценок. Очевидно, что чем больше фильм соответствует интересам пользователя, тем выше оценивается потенциальная заинтересованность пользователя.

User based рекомендации, алгоритм рекомендаций который используется в разработке ресурса RandMovie. Это такие рекомендации, в рамках которых, рекомендации генерируются на основе интересов похожих пользователей. Такие рекомендации являются результатом коллаборации множества пользователей, которая основана на принципе к ближайших соседей. То есть для каждого из пользователей ресурса происходит поиск в системе к наиболее похожих на него дополняя информацию о пользователе известными данными по его «соседям». Слово похожесть является в данном случае синоним корреляции интересов и может считаться множеством способов. Все остальные алгоритмы строятся на основе уже перечисленных и является их модификациями, а также их бесчисленное множество.

Так как каждый пользователь оценивает по-своему, кто-либо подряд ставит десятки, а кто-то в плохом настроении ставит одну или две звезды, перед расчетом данные необходимо привести к единой шкале для того, чтобы выбранный алгоритм мог корректно сравнивать пользователей между собой. Похожесть или корреляцию предпочтений двух пользователей можно считать разными способами. По сути, необходимо сравнить два вектора и сделать это можно различными способами, например с помощью корреляции Пирсона, который является классическим коэффициентом. Корреляция Спирмана, в отличие от корреляции Пирсона, данный коэффициент не работает с абсолютными значениями рейтингов, но работает с их порядковыми номерами. Косинусное расстояние является еще одним классическим коэффициентом, косинусное, потому что, если два вектора сонаправлены (т.е. угол между ними нулевой), то косинус угла между ними равен единице. И наоборот, косинус угла между перпендикулярными векторами равен нулю.

Очень важно, чтобы пользователь доверял рекомендательной системе, а для этого она должна быть проста и понятна. При необходимости всегда должно быть доступно понятное объяснение рекомендации. В рамках объяснения неплохо показывать оценку фильма соседями, по какому именно атрибуту (например, актер или режиссер) было совпадение, а также выводить уверенность системы в оценке.

Список используемых источников:

1. Анатомия рекомендательных систем. Часть первая [Электронный ресурс]. – 2018 – Режим доступа: <https://habr.com/ru/company/lanit/blog/420499/> дата обращения – 04.02.2020.
2. Анатомия рекомендательных систем. Часть вторая [Электронный ресурс]. – 2018 – Режим доступа: <https://habr.com/ru/company/lanit/blog/421401/> дата обращения – 07.02.2020.

ОСОБЕННОСТИ БАНКОВСКОЙ СИСТЕМЫ США

Черная К.В., студент группы 17Г91, научный руководитель: Лощилова М.А.

Юргинский технологический институт (филиал) Национального исследовательского

Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

E-mail: marisha20@bk.ru

Аннотация: В статье рассматриваются важнейшие моменты в истории банковской системы США, изменения в структуре американской банковской системе на разных этапах ее развития, особенности статуса национальных банков и банков штатов. Делается вывод о том, что своеобразие современной банковской системы США во многом определяется ее историей.

Ключевые слова: банковская система США

Банковская система США стала формироваться в период войны за независимость (1775 - 1783 гг.), которая обошлась в 100 млн. долларов. Перед Континентальным конгрессом встала весьма сложная задача финансирования войны. Собственная банковская система в колониях отсутствовала, а использование финансовых каналов, связанных с банками Великобритании, стало затруднительным.