

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Прикладные информационные системы (ПИС-2022)

Сборник научных трудов
VIII Всероссийской научно-практической конференции
с международным участием

(г. Ульяновск, 30 мая – 05 июня 2022 г.)

Ульяновск
УлГТУ
2022

УДК 004.9(082)
ББК 32.973–018.2я43
П 75

Редакционная коллегия: Н.Г. Ярушкина,
А.А. Романов,
А.А. Филиппов,
Е.Н. Эгов (ответственный за выпуск)

УДК 004.9(082)

Прикладные информационные системы (ПИС-2022) :
сборник научных трудов VIII Всероссийской научно-практической
конференции с международным участием (г. Ульяновск, 30 мая –
05 июня 2022 г.) [Электронный ресурс]. – Электронные текстовые
данные. Ульяновск: УлГТУ, 2022. – 91 с.

В сборнике опубликованы доклады участников VIII Всероссийской
научно-практической конференции «Прикладные информационные системы
(ПИС-2022)».

Материалы сборника предназначены для специалистов по
информационным системам и технологиям, магистрантам и аспирантам
информационно-технических специальностей вузов.

Статьи представлены в авторской редакции.

ISBN 978-5-9795-2245-6

© Коллектив авторов, 2022
© Оформление, УлГТУ, 2022

Оргкомитет ПИС–2022:

Председатель:

Ярушкина Н.Г., д.т.н., проф., УлГТУ, г. Ульяновск

Члены организационного комитета:

Романов А.А., к.т.н., доцент, УлГТУ, г. Ульяновск

Филиппов А.А., к.т.н., УлГТУ, г. Ульяновск

Эгов Е.Н., УлГТУ, г. Ульяновск

Скалкин А.М., УлГТУ, г. Ульяновск

Строева Ю.В., УлГТУ, г. Ульяновск

Программный комитет ПИС–2022:

Председатель:

Ярушкина Н.Г., д.т.н., проф., УлГТУ, г. Ульяновск

Заместитель Председателя:

Наместников А.М., д.т.н., доцент, УлГТУ, г. Ульяновск

Члены программного комитета:

Гуськов Г.Ю., к.т.н., УлГТУ, г. Ульяновск

Мошкин В.С., к.т.н., УлГТУ, г. Ульяновск

Мошкина И.А., к.т.н., УлГТУ, г. Ульяновск

Романов А.А., к.т.н., доцент, УлГТУ, г. Ульяновск

Филиппов А.А., к.т.н., УлГТУ, г. Ульяновск

*VIII Всероссийская научно-практическая конференция
с международным участием
Прикладные информационные системы-2022*

VIII Всероссийская научно-практическая конференция с международным участием «Прикладные информационные системы-2022» проведена на базе Ульяновского государственного технического университета.

В рамках конференции было представлено пятнадцать научных докладов по следующим тематикам: «Интеллектуальный анализ данных», «Инженерия знаний, онтологий, управления знаниями», «Нечеткие системы и мягкие вычисления», «Проектирование информационных систем». Также в рамках конференции была проведена Молодежная школа-семинар, в которой участвовали бакалавры и магистранты технических специальностей.

Организационный комитет благодарит авторов докладов, приславших свои работы на конференцию, а также ректорат Ульяновского государственного технического университета, обеспечивший проведение конференции и издание ее материалов.

Председатель
организационного комитета конференции доктор
технических наук, профессор

Н.Г. Ярушкина

СОДЕРЖАНИЕ

<i>Гладилин Н.Д.</i> Разработка сервиса рекомендаций медиа контента	10
<i>Горяйнов А.В., Гуськов Г.Ю.</i> Разработка информационной системы «платформа подбора участников медийных мероприятий».....	14
<i>Костин Д.А.</i> Разработка web-сервиса формирования профилей научных сотрудников на языке Python	21
<i>Лебедев Д.О.</i> Разработка образовательного портала для центра цифрового образования «It-Cube» на фреймворке Django	24
<i>Маринова В.В.</i> Разработка сайта для поиска удаленной работы	28
<i>Охотников Д.В., Гуськов Г.Ю.</i> Разработка информационной системы «Web-приложение для торговли лимитированными товарами»	31
<i>Пиков К.А.</i> Разработка виджета для автоматизации процессов малого бизнеса в сфере продаж и оказания услуг. Слой интеграции.....	37
<i>Смирнов А.А.</i> Разработка системы регистрации для ролевых онлайн-игр на веб-фреймворке Laravel с интеграцией в Discord и Mediawiki	44
<i>Сяман С.В.</i> Разработка многопользовательской мобильной игры в жанре «Стратегия»	52
<i>Тягин И.С.</i> Разработка сервиса рекомендаций компьютерных игр на языке Python	56

<i>Утицких Д.А.</i> Разработка автоматизированной системы для бухгалтерского учета в малом бизнесе на .Net Framework	59
<i>Фаворская М.Н., Кутузова М.О.</i> Обнаружение презентационных атак при распознавании лиц на основе детектирования движения век	62
<i>Хоменко Г.К.</i> Разработка интеллектуальной системы по подбору комплектующих для ПК.....	68
<i>Черноморченко А.В., Романов А.А.</i> Разработка приложения «Информационный портал об инвестициях».....	72
<i>Шпанцев А.А., Гуськов Г.Ю.</i> Разработка информационной системы «web-приложение торговой площадки для лимитированных товаров на основе интернет-аукциона»	82
Авторский указатель.....	91

CONTENTS

<i>Gladilin N.D.</i> Development of media content concentration service	10
<i>Goryaynov A.V., Guskov G.Y.</i> Development of the information system «Media events castings platform»	14
<i>Kostin D.A.</i> Development of a web service for the formation of a profile of researchers in the Python language.....	21
<i>Lebedev D.O.</i> Development of an educational portal for the digital education center "It-Cube" using Django framework	24
<i>Marinova V.V.</i> Development of a website for remote job search.....	28
<i>Okhotnikov D.V., Guskov G.Yu.</i> Development of information system «Web-application for trade in limited edition goods»....	31
<i>Pikov K.A.</i> Development of a widget for automating small business processes in the field of sales and services. Integration layer.....	37
<i>Smirnov A.A.</i> Development of a registration system for roleplay online games on the Laravel web-framework with integration into Discord and Mediawiki	44
<i>Syaman S.V.</i> Development of a multiplayer mobile game in the «Strategy» genre	52
<i>Tyagin I.S.</i> Development of a computer game recommendation service in Python.....	56
<i>Utitskikh D.A.</i> Development of an automated system for accounting in a small business on the .Net Framework	59
<i>Favorskaya M.N., Kutuzova M.O.</i> Detection of presentation attacks in face recognition based on eyelid movement	62

<i>Khomenko G.K.</i> Development of media content concentration service	68
<i>Chernomorchenko A.V., Romanov A.A.</i> Development of the application «Investment information portal»	72
<i>Shpantsev A.A., Guskov G.Y.</i> Development of information system «Web-application of the marketplace for limited products on the basis of the internet auction»	82
Authors index	91

УДК 004.94

РАЗРАБОТКА СЕРВИСА РЕКОМЕНДАЦИЙ МЕДИА КОНТЕНТА

Гладилин Н.Д. (splint38@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация сервиса рекомендаций с возможностью предоставления алгоритма рекомендаций к использованию через API. Производится сравнение с аналогами в сфере рекомендаций.

Ключевые слова: рекомендация, датасет, контент, данные, алгоритм, API.

Введение

В настоящее время пользователи заинтересованы в получении релевантных рекомендаций для экономии времени при работе с товарами, услугами или контентом. На данный момент многие компании внедряют сервисы рекомендаций в свои проекты, такие как веб-сайты или мобильные приложения для предоставления пользователю альтернатив.

Современные сервисы рекомендаций обладают обширным функционалом от отправки триггерных сообщений на почту пользователя до составления рекомендации по изображению товара. Но в большинстве случаев данные сервисы тяжело интегрируются в систему, требуют большой объём данных для работы и обладают высокой стоимостью использования.

Таким образом было принято решение разработать сервис рекомендаций с возможностью простой и гибкой интеграции, с небольшой стоимостью и с меньшим объёмом начальных данных для рекомендации.

Анализ существующих решений

Рассмотрим существующие на данный момент сервисы рекомендаций. Самыми популярными на данный момент являются Rees46, система рекомендаций Amazon.

Rees46 предоставляет пользователю товарные рекомендации, которые формируются на основе большого количества данных. Сервису необходимо предоставлять огромное количество информации от действий пользователя до детального описания товаров. Система внедряется достаточно сложно через товарную выгрузку XML, а инструкция поддерживает только 3 языка программирования. Сервис обладает демоверсией для просмотра функционала. Цена вопроса составляет от 5000 рублей в месяц.

Таким образом Rees46, подходит только крупным компаниям так как для работы необходим большой объём данных, и сервис имеет большую стоимость внедрения. Сложность внедрения так же подразумевает наличие специалистов высокого уровня. Система получается недостаточно гибкой так как постоянно упирается в действия пользователя.

Система рекомендаций Amazon позволяет использовать те же технологии машинного обучения, которые использует Amazon.com. Обладает широким спектром возможностей таких как рекомендации конкретных продуктов, сортировку продуктов по персонализированному рейтингу и индивидуальный прямой маркетинг. Система предлагает доступ к алгоритмам либо через специальное приложение Amazon SageMaket, либо через api, поддерживает много языков программирования, но обладает сложным процессом внедрения сложных вычислений. Сервис сложен в использовании так как обладает огромным количеством настроек, в которых необходимо разбираться. Демо версия не предусмотрена Цена вопроса от 10 центов до 5 долларов в час.

Можно сказать, что система Amazon это ультимативный вариант рекомендационной системы, который предоставляет гибкие настройки и обладает огромным количеством алгоритмов, но за все это приходится платить очень высокой стоимостью и сложностью интеграции.

Предлагаемое решение

Таким образом было принято решение разработать доступный по стоимости, гибкий и легкий в интеграции сервис рекомендаций. Для демонстрации работы которого предоставляется рекомендация медиа контента в трех сферах: кино, сериалы, игры.

Для реализации проекта было принято решение использовать фреймворк Django так как он отлично подходит для сайтов-сервисов, а также у него имеется api расширение.

Целью данного проекта является система, основанная на нейронном алгоритме построения TFIDF матрицы для рекомендации медиа контента с возможностью получить рекомендацию по датасету.

Конечная система представляет собой веб-сайт с возможностью получить рекомендацию в сфере игр, кино или сериалов. Пользователь выбирает интересующую его сферу контента и вводит название контента, на основе которого он хочет получить рекомендацию (например, понравившийся фильм), и система выдаст пользователю список рекомендованного контента. Пользователь имеет возможность зарегистрироваться для того, чтобы добавлять контент в медиатеку (контент в медиатеке не будет отображаться в рекомендации), а также загружать собственные наборы данных с возможностью применения алгоритма и получения рекомендации по загруженному датасету.

Так же сервис предоставляет API для желающих воспользоваться алгоритмом рекомендации. Через API пользователь загружает датасет (характеристики которого описаны на странице информации) сервис сохраняет модель для работы с датасетом. После чего пользователь может получить рекомендацию по-своему набору данных и получить результат в формате json. Для получения рекомендации через API достаточно лишь обратиться по ссылке и получить json ответ от сервиса.

Система будет разворачиваться на собственном аппаратном обеспечении. Цена вопроса 600 рублей в месяц.

Заключение

В качестве итога работы был разработан сервис рекомендаций медиа контента. Сервис обладает легким способом внедрения и достаточной гибкостью, так как может работать с загружаемыми пользователем датасетами. Для работы с сервисом нет необходимости разбираться с огромным числом настроек или передавать огромный объем данных, достаточно загрузить датасет по шаблону. Стоимость намного меньше, чем у конкурентов.

Таким образом все поставленные задачи были выполнены, но сервис может быть подвергнут расширению, например, добавлением разных алгоритмов рекомендации на выбор. Добавление алгоритмов с возможностью производить прогнозирование на основе выбранного пользователем контента.

Список литературы

1. Amazon Personalize [Электронный ресурс] – Режим доступа: <https://aws.amazon.com/ru/personalize/>
2. Amazon SageMaker [Электронный ресурс] – Режим доступа: <https://aws.amazon.com/ru/sagemaker/>

3. Django framework [Электронный ресурс] – Режим доступа: <https://www.djangoproject.com/>
4. REES46 [Электронный ресурс] – Режим доступа <https://rees46.ru/>
5. Обзор сервисов персональных рекомендаций [Электронный ресурс] – Режим доступа: <https://blog.sibirix.ru/recommender-system-review/>

DEVELOPMENT OF MEDIA CONTENT CONCENTRATION SERVICE

Gladilin N.D. (splint38@gmail.com)
Ulyanovsk state technical University, Ulyanovsk

This article describes the implementation of a recommendation service with the ability to provide a recommendation algorithm for use via api. A comparison is made with analogues in the field of recommendations.

Keywords: recommendation, dataset, content, data, algorithm, api.

УДК 004.415

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ «ПЛАТФОРМА ПОДБОРА УЧАСТНИКОВ МЕДИЙНЫХ МЕРОПРИЯТИЙ»

Горяйнов А.В. (gor.alex595@gmail.com)

Гуськов Г. Ю. (guskovgleb@gmail.com)

Ульяновский Государственный Технический
Университет, Ульяновск

В работе представлено описание разработанной информационной системы «Платформа подбора участников медийных мероприятий». Система предназначена для автоматизации сбора заявок участников телевизионных проектов, стандартизации формата заявок и упрощения процесса вынесения решений по ним для представителей телеканалов.

Ключевые слова: медиа, приём заявок, телевизионные проекты, подбор участников, Laravel, Angular, Data Access Layer

Введение

В настоящее время общество переходит на постиндустриальную стадию развития, что характеризуется стремительным ростом объёмов информации, в том числе и в сфере масс-медиа и телевидения. С каждым годом обновляется формат вещания телевизионных систем, появляется всё больше телеканалов, запускаются новые медийные проекты. Телеканалам, находящимся в условиях конкуренции за просмотры и активность пользователей, приходится более тщательно подходить к формированию контента, образа телеканала и искать наименее ресурсозатратные способы проведения отбора участников медийных мероприятий.

Требуется разработать платформу, позволяющую автоматизировать процесс сбора заявок участников, стандартизировать формат заявок и упростить их систематизацию, просмотр и процесс вынесения решения для представителей телеканалов.

Основные бизнес-процессы платформы:

- Объявление набора участников на проекты;

- Сбор заявок участников проектов;
- Систематизация и просмотр заявок;
- Вынесение решения по заявке.

Функциональные требования, предъявляемые к системе:

- Регистрация и авторизация в системе (Регистрация пользователя возможна в роли участника проектов или менеджера телеканалов. Авторизация приложения должна определять роль участника по его адресу электронной почты и выдавать токен доступа с соответствующими правами (участник, менеджер, администратор системы));
- Управление проектами (менеджер телеканала должен иметь возможность создать новый проект, открыв для него набор участников, и редактировать существующий проект);
- Каталоги телеканалов, проектов, заявок (все участники проектов должны иметь возможность просмотреть каталог телеканалов и проектов, на которые открыт набор. Платформа должна предоставлять участникам и менеджерам возможность просмотра заявок, связанных с их деятельностью в системе)
- Управление заявками (участник должен иметь возможность подать заявку на понравившийся проект, а в случае непредвиденных обстоятельств - отменить её. В интерфейсе менеджера телеканала должны присутствовать функции одобрения или отклонения заявок участников).
- Просмотр статистики активности (администратору должна быть предоставлена возможность просмотреть статистику активности пользователей на сайте для оценки возможных направлений дальнейшего развития платформы).

Проектирование системы

Платформа представляет собой клиент-серверное приложение. Бэкенд-приложение написано на языке программирования PHP (фреймворк Laravel), фронтенд – на JavaScript-фреймворке Angular. Для хранения данных используются СУБД MySQL и система кэширования Redis. Клиент взаимодействует с сервером путём обращения к методам его API.

Веб-приложения организовано с опорой на архитектурный паттерн Model-View-Controller. При проектировании бэкенд-части активно применялись паттерн Data Access Layer и внедрение зависимостей.

В результате анализа ключевых прецедентов системы было выявлено 12 вариантов использования и 4 актора (неавторизованный пользователь, участник проекта, менеджер телеканала и администратор системы).

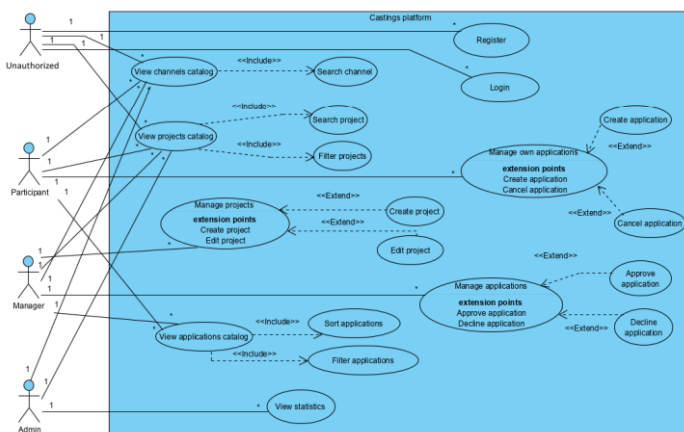


Рисунок 1 – Диаграмма вариантов использования

Разработанные приложения должны быть развёрнуты на локальном веб-сервере или хостинге, доступном для обращения по HTTP-протоколу.

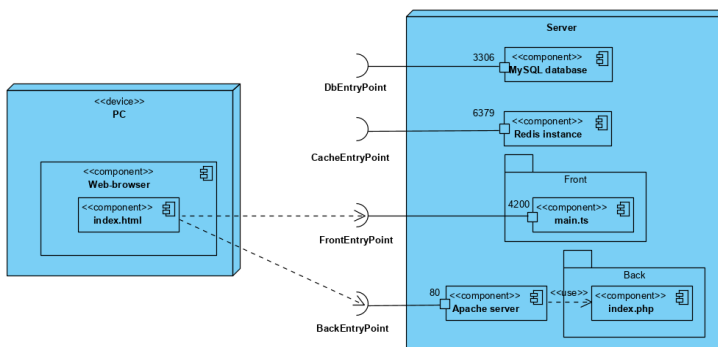


Рисунок 2 – Диаграмма развёртывания системы

Программная реализация системы

На странице каталога телеканалов отображаются карточки телеканалов, зарегистрированных в системе, содержащие их фотографии и описание, кроме того доступен поиск по названию телеканала.

Каталог проектов отображает мероприятия, на которые открыт набор участников, с подробной информацией о месте их проведения, дате начала, гонораре и лимите участников (рисунок 3). В каталоге реализованы фильтры по телеканалу, городу, наличию мест и гонорара за участие.

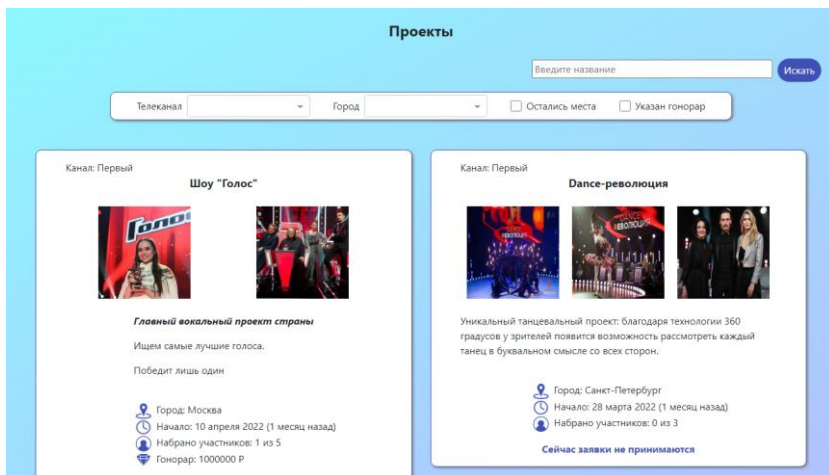


Рисунок 3 – Каталог проектов

Пользователь, авторизованный в роли менеджера телеканала, может создавать проекты для собственного телеканала, открывая набор участников, а также редактировать проекты, изменяя описание проекта, место и дату его проведения и требуемое число участников, что позволяет более гибко управлять процессом подбора кандидатов.

После регистрации участнику становится доступна возможность подачи заявки на доступные проекты. Создание заявки подразумевает написание текстового мини-резюме, описывающего таланты участника и цели его участия в проекте (задачу форматирования текста резюме упрощает встроенный WYSIWYG-редактор CkEditor), а также прикрепление видеофайла, записав который, участник проекта может дополнительно продемонстрировать свои навыки. (рисунок 4).

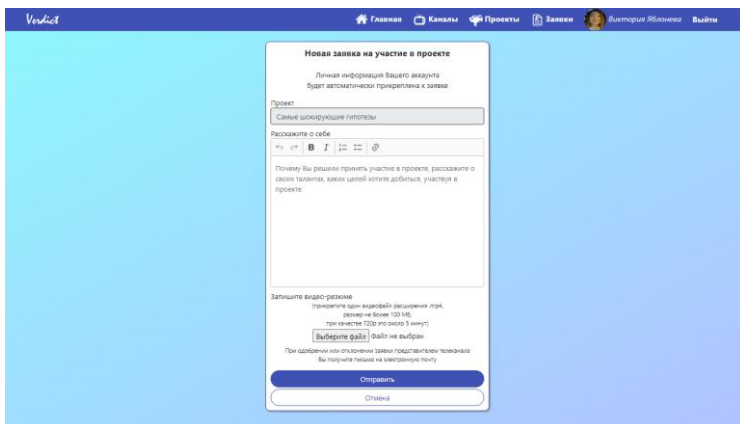


Рисунок 4 – Форма создания заявки на проект

Поданные заявки отображаются как в аккаунте участника, так и в кабинете менеджера соответствующего телеканала в каталоге заявок (рисунок 5)

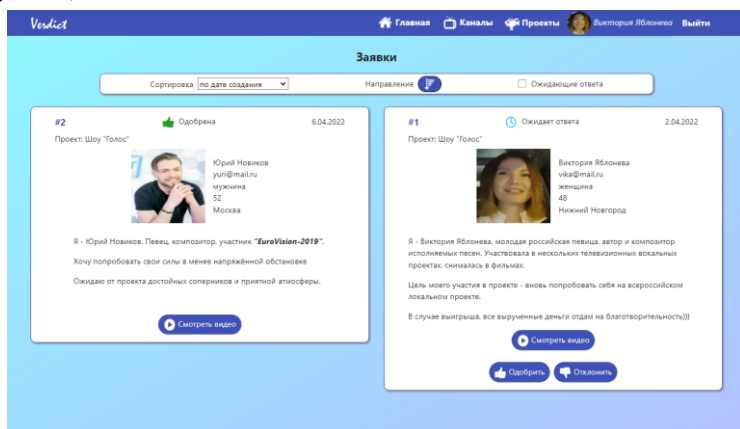


Рисунок 5 – Каталог заявок

Администратору системы доступна статистика активности участников платформы, помогающая оценить популярность отдельных телеканалов и выбрать направление дальнейшего развития платформы.

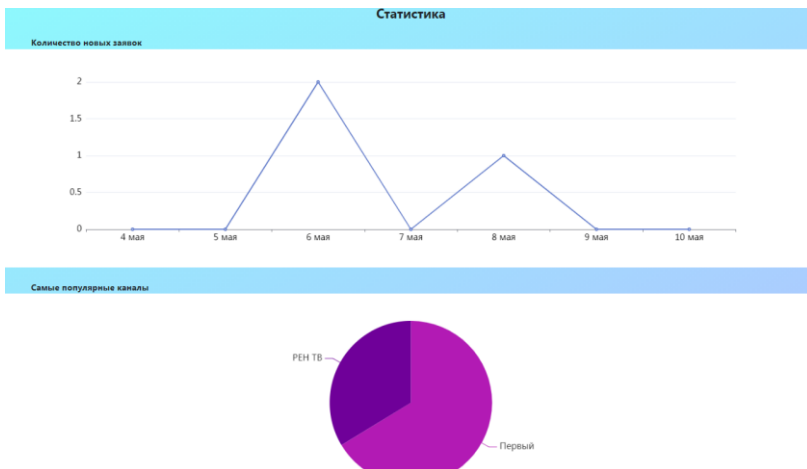


Рисунок 6 – Статистика в панели администратора

Заключение

Результатом работы является полностью работоспособное клиент-серверное приложение «Платформа подбора участников медийных мероприятий», полностью отвечающее поставленным требованиям.

Разработанное приложение позволит сократить время, затрачиваемое менеджерами телеканалов на монотонную работу, и сделает процесс прохождения кастингов более приятным и простым для участников проектов. Кроме того, дистанционный формат работы платформы подбора участников станет несомненным преимуществом при проведении кастингов на всей территории страны, что значительно упростит процесс для участников, проживающих в городах, удалённых от местоположения организации, проводящей кастинги.

В будущем планируется расширение функциональности платформы. Например, введение стадии голосования за одобрение заявки участника среди нескольких менеджеров телеканалов.

Список литературы

1. Корунова Н.В. Проектирование программного обеспечения: применение на практике / практикум по дисциплине «Проектирование информационных систем» / Н. В. Корунова. – Ульяновск: УлГТУ, 2019. – 69 с. - URL: <http://is.ulstu.ru/disc/pris>. — Режим доступа: для авториз. пользователей.
2. Фримен А., Angular для профессионалов. — СПб.: Питер, 2018. — 800 с.

3. Зандстра М., PHP: объекты, шаблоны и методики программирования, 4-е изд.: Пер. с англ. – М.: ООО "И.Д. Вильямс", 2015. – 576 с.: ил. – Парал. тит. англ.

DEVELOPMENT OF THE INFORMATION SYSTEM «MEDIA EVENTS CASTINGS PLATFORM»

Goryaynov A.V. (gor.alex595@gmail.com)

Guskov G.Y. (guskovgleb@gmail.com)

Ulyanovsk state technical University, Ulyanovsk

The paper presents a description of the developed information system: "Media events castings platform", which supposed to automate the process of uploading projects' participants applications, to standardize their format and to simplify the process of making decisions on applications for managers of TV channels.

Keywords: media, application uploading, TV project, casting, Laravel, Angular, Data Access Layer

УДК 004.94

РАЗРАБОТКА WEB-СЕРВИСА ФОРМИРОВАНИЯ ПРОФИЛЕЙ НАУЧНЫХ СОТРУДНИКОВ НА ЯЗЫКЕ PYTHON

Костин Д.А. (wolf.1998.gig@yandex.ru)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация сервиса формирования профиля научного сотрудника с возможностью расширения функционала с помощью модулей по сбору данных. Производится сравнение с аналогами в сфере.

Ключевые слова: сбор, данные, наука, статья, модуль.

Введение

В наше время ежегодно происходит 415 тысяч научных исследований который охватывают огромное количество областей от формы снежинок до квазаров. Сейчас тысячи ученых проводят исследования, которым иногда посвящают года, десятилетия или даже всю жизнь.

Как в химии существует понятие катализатор, которое означает вещество, ускоряющее химическую реакцию, так и в науке можно ускорить любое исследование с помощью людских ресурсов и финансирования.

Для каждого человека, посвятившего жизнь науке довольно время затратно разбираться в области составлении бумаг и документов, а еще сложнее, следить за всеми своими публикациями и различными отметками в научных работах.

Таким образом было принято решение разработать сервис, который будет собирать информацию с нескольких источников и формировать в единый научный профиль.

Анализ существующих решений

Рассмотрим существующие на данный момент сервисы по сбору и индексированию информации о научной деятельности. Самыми популярными на данный момент являются Elibrary и Scopus.

eLIBRARY.RU – крупнейшая в России электронная библиотека научных публикаций, обладающая богатыми возможностями поиска и анализа научной информации. Библиотека интегрирована с Российским индексом научного цитирования (РИНЦ) – созданным по заказу Минобрнауки РФ бесплатным общедоступным инструментом измерения публикационной активности ученых и организаций. eLIBRARY.RU и РИНЦ разработаны и поддерживаются компанией «Научная электронная библиотека».

Scopus. Для авторов, которые опубликовали более одной статьи, в Scopus создаются индивидуальные учётные записи – профили авторов с уникальными идентификаторами авторов (Author ID). Эти профили предоставляют такую информацию, как варианты имени автора, перечень мест его работы, количество публикаций, годы публикационной активности, области исследований, ссылки на основных соавторов, общее число цитирований на публикации автора, общее количество источников, на которые ссылается автор, индекс Хирша автора и т. д. База данных предоставляет пользователям возможности использования уникальных идентификаторов авторов для формирования поисковых запросов и настройки оповещений электронной почтой или через RSS) по изменениям в профилях авторов.

Возможности поиска авторов и ограниченного просмотра их профилей доступны без абонентской подписки на базу данных Scopus средствами Scopus Author Preview.

Предлагаемое решение

Таким образом было принято решение разработать доступный по стоимости, расширяемый и легкий в использовании сервис по формированию профиля научных сотрудников. Для демонстрации работы которого представляется сбор данных с сайта elibrary.ru.

В процессе разработки было принято решение использовать фрейм ворк Django так как наш сервис должен быть достаточно гибким для дальнейшего расширения, а Django сейчас очень популярен поэтому будет не сложно найти людей для модифицирования проекта.

Целью является система по сбору информации с различных модулей в приложении и вывод на экран пользователя профиль научного сотрудника. В системе имеется стандартный модуль, который собирает данные с Elibrary.

Конечная система представляет собой веб-сайт который позволяет пользователю получить информацию о любом научном сотруднике по имени. Пользователь может зарегистрироваться и дополнить информацией профиль любого научного сотрудника.

Конечная система гибко расширяется если поместить новый модуль в папку. Система будет разворачиваться на собственном аппаратном обеспечении. Цена вопроса 249 рублей в месяц.

Заключение

В качестве итога работы был разработан сервис по сбору информации и формированию профиля научного сотрудника. Сервис обладает специальной системой для гибкого расширения источников информации. Для работы с сервисом не нужны никакие специальные знания кроме базовых по использованию компьютера. Стоимость намного меньше, чем у конкурентов.

Таким образом все поставленные задачи выполнены, но сервис может быть расширен для использования дополнительных источников информации о научной деятельности.

Список литературы

1. Elibrary [Электронный ресурс] – Режим доступа: <https://www.elibrary.ru/defaultx.asp>
2. Scopus Preview [Электронный ресурс] – Режим доступа: <https://www.scopus.com/home.uri>
3. Django framework [Электронный ресурс] – Режим доступа: <https://www.djangoproject.com/>
4. Selenium automates browsers [Электронный ресурс] – Режим доступа: <https://www.selenium.dev/>

DEVELOPMENT OF A WEB SERVICE FOR THE FORMATION OF A PROFILE OF RESEARCHERS IN THE PYTHON LANGUAGE

Kostin D.A. (wolf.1998.gig@yandex.ru)
Ulyanovsk state technical University, Ulyanovsk

This article describes the implementation of the researcher profile formation service with the possibility of expanding the functionality using data collection modules. A comparison is made with analogues in the field.

Keywords: collection, data, science, article, module.

УДК 004.94

РАЗРАБОТКА ОБРАЗОВАТЕЛЬНОГО ПОРТАЛА ДЛЯ ЦЕНТРА ЦИФРОВОГО ОБРАЗОВАНИЯ «IT-CUBE» НА ФРЕЙМВОРКЕ DJANGO

Лебедев Д.О. (inzadimonax@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация цифрового образовательного портала с возможностью создания курсов и автоматической проверкой задач. Производится сравнение с аналогами в сфере рекомендаций.

Ключевые слова: образовательный портал, курсы, контент, данные, задача по программированию.

Введение

Центр образования детей «IT-Куб» работает по программам, направленным на ускоренное освоение актуальных и востребованных знаний, навыков и компетенций в сфере информационных технологий. Программирование является неотъемлемой частью сферы информационных технологий. Оно изучается на направлении «Программирование на Python».

По ходу обучения требуется изучить методы программирования на языке Python; рассмотреть различные парадигмы программирования, предлагаемые этим языком (процедурная, функциональная, объектно-ориентированная); подготовить обучающихся к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях. Обучение производится на основе изучения обучающимися лекционного материала и решения задач по программированию. Преподаватель читает лекцию и даёт обучающимся задачи по программированию для решения, а потом проверяет решение.

Требуется разработать информационную систему (ИС), позволяющую автоматизировать проверку решения задач по программированию, структурировать курсы обучения, автоматизировать учёт успеваемости.

Анализ существующих решений

Рассмотрим существующие на данный момент образовательные платформы. Самыми популярными на данный момент являются: система управления курсами Moodle, образовательная платформа Stepik.

Stepik – российская образовательная платформа и конструктор бесплатных и платных открытых онлайн-курсов, и уроков. Она позволяет любому зарегистрированному пользователю создавать интерактивные обучающие уроки и онлайн-курсы, используя видео, тексты и разнообразные задачи с автоматической проверкой и моментальной обратной связью. В процессе обучения студенты могут вести обсуждения между собой и задавать вопросы преподавателю на форуме. Автоматическая проверка осуществляется на языке Python версии 3.6. Также на платформе отсутствуют сборники задач, которые независимы от курсов. Стоимость для образовательных учреждений составляет от 500000 рублей в год.

Таким образом Stepik, резюмируя, стоимость для организации высока. На Stepik нет привязки задач к теории, а также нет сборников задач. Также на портале отсутствует песочница, с помощью которой возможно запускать код. Сложность внедрения так же подразумевает наличие специалистов высокого уровня.

Moodle – система управления курсами (электронное обучение), также известная как система управления обучением или виртуальная обучающая среда. Представляет собой свободное (распространяющееся по лицензии GNU GPL) веб-приложение, предоставляющее возможность создавать сайты для онлайн-обучения. Платформа предоставляет пространство для совместной работы учителей и студентов. В Moodle доступны различные возможности для отслеживания успеваемости учащихся, а также есть поддержка массовой регистрации с безопасной аутентификацией. Платформу можно интегрировать с большим количеством программного обеспечения, включая инструменты для общения, совместной работы, управления документами и другие приложения для повышения производительности. Moodle имеет открытый исходный код.

В Moodle есть привязка задач к теории, но нет сборников задач, также отсутствует песочница, с помощью которой возможно запускать код. Система управления курсами является очень тяжеловесной и для работы в ней, нужно обучать и преподавателей и обучающихся, а ещё для создания хорошего курса от преподавателя требуются технические компетенции в области веб-разработки. Так как обучающимися являются школьники, Moodle не подходит из-за излишней сложности интерфейса и отсутствия песочницы для программирования.

Анализ даталиб, насколько лучше аналогов, насколько лучше решает задачи пользователей.

Предлагаемое решение

Таким образом было принято решение разработать доступный по стоимости, гибкий и легкий цифровой образовательный портал.

Для реализации проекта было принято решение использовать фреймворк Django так как он отлично подходит для сайтов-платформ.

Преподаватель загружает лекционные материалы и задачи в ИС. Рассказывает теорию по теме. Обучающиеся просматривают лекционные материалы и код по ним, внося изменения в пример кода, тем самым понимая работу алгоритма. Затем обучающиеся решают задачи в ИС. ИС проверяет решение и выдаёт ошибку, из-за которой задача решена неверна, либо принимает решение задачи. ИС проверяет код задач на языке программирования Python 3.8, а аналоги используют Python 3.6.

Итогом системы является то, что обучающиеся будут лучше воспринимать информацию и понимать алгоритмы кода, соответственно повысится качество обучения. Преподавателю не нужно вручную проверять задачи.

Система будет разворачиваться на собственном аппаратном обеспечении. Стоимость содержания оборудования 1000 рублей в месяц.

Заключение

В качестве итога работы был разработан цифровой образовательный портал. В портале реализована система курсов, которая включает в себя систему проверки задач, разбиение курса на модули, а модули разбиваются на блоки контента. Контентом может являться: текст, видео, картинка, файл или песочница с кодом, также к контенту можно прикрепить задачу, которая автоматически проверится. Песочница с кодом отсутствует у аналогов, также в аналогах невозможно прикрепить задачу к контенту. Расположение всех функций расположено интеллектуально понятно и у пользователей не возникнет вопросов, где находится та или иная функция. Стоимость намного меньше, чем у конкурентов.

Таким образом все поставленные задачи были выполнены, но портал может быть подвергнут расширению, например, добавление других языков программирования для проверки, а также обширной статистики для курсов.

Список литературы

1. Stepik [Электронный ресурс] – Режим доступа: <https://stepik.org/>
2. Moodle [Электронный ресурс] – Режим доступа: <https://moodle.org/>
3. Django framework [Электронный ресурс] – Режим доступа: <https://www.djangoproject.com/>

DEVELOPMENT OF AN EDUCATIONAL PORTAL FOR THE DIGITAL EDUCATION CENTER "IT-CUBE" USING DJANGO FRAMEWORK

Lebedev D.O. (inzadimonaxgmail.com)
Ulyanovsk state technical University, Ulyanovsk

This article describes the implementation of a digital educational portal with the ability to create courses and automatically check tasks. A comparison is made with analogues in the field of recommendations.

Keywords: educational portal, courses, content, data, programming task.

УДК 004.94

РАЗРАБОТКА САЙТА ДЛЯ ПОИСКА УДАЛЕННОЙ РАБОТЫ

Маринова В.В. (valeria.marinova.00@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация сайта для поиска удаленной работы с автоматическим подбором кандидата на вакансию. Кандидат подбирается с учетом его рейтинга, стажа работы и специализации.

Ключевые слова: рекомендация, датасет, специализация, подбор, данные, алгоритм.

Введение

В настоящее время многие люди не имеют возможности работать в офисе. На это может быть много причин, например, проблемы со здоровьем, семейные обстоятельства, невозможность сменить место жительства и многое другое. Для решения данной проблемы люди вынуждены жертвовать какими-то определенными ресурсами или же рискуют просто остаться без работы и заработной платы.

Таким образом было принято решение разработать сайт для удаленной работы. Основной задачей которого является поиск работы для людей по различным специализациям, задачи предоставляют заказчики. Существует разделение по правам доступа у исполнителя и заказчика. Заказчик, выбрав специализацию предоставляет вакансию, к его заказу подбираются рекомендуемые исполнители, исполнители видят рекомендуемое предложение по работе и либо соглашаются, либо отказываются от предоставленной вакансии, так же имеется чат для того, чтобы заказчик и исполнитель могли обсудить детали заказа.

Анализ существующих решений

Рассмотрим существующие на данный момент сайты для поиска удаленной работы. Самым популярным на данный момент сайтом для поиска удаленной работы является FL.ru-сервис для поиска удаленной

работы и сотрудников в России и СНГ.

FL.ru позволяет осуществлять вход на сайт в качестве заказчика и публиковать свой заказ. Исполнитель же в свою очередь видит список появившихся заказов и может при желании откликнуться на любой, интересующий его заказ. На сайте предусмотрена оплата, цена безопасной оплаты – от 1200, что бывает несоизмеримо с ценой заказа, помимо этого, существует небезопасная оплата, где пользователь вполне может столкнуться с мошенничеством. Интерфейс сайта так же имеет недостатки на странице предлагаемых вакансий, заключающиеся в том, что оставленная заявка на подбор исполнителя зачастую может потеряться в потоке других предложений.

Проанализировав одно из существующих решений, можно сказать, что сайт для удаленной работы FL.ru является довольно неплохим решением для поиска удаленной работы и подбора кандидата на вакансию, но также имеет недостатки в виде платной безопасной сделки и недостаточно удобного интерфейса.

Предлагаемое решение

Таким образом было принято решение разработать сайт для поиска удаленной работы с автоматическим подбором кандидата на вакансию с понятным интерфейсом и без взимания дополнительной платы.

Для реализации проекта было принято решение использовать фреймворк Django так как он отлично подходит для реализации алгоритма подбора кандидата на задачу.

Конечная система представляет собой веб-сайт где исполнитель создает вакансию и на нее автоматически подбираются кандидаты, которые делятся на наиболее подходящего и остальных кандидатов по направлениям, специализации и рейтингу, в зависимости от стоимости и занятости исполнителя, заказчик выбирает кандидата и при необходимости может начать с ним диалог в чате для обсуждения дополнительных деталей заказа. По мере договоренности и получения результата заказа, исполнитель может поставить оценку исполнителю, из полученных оценок формируется среднее для дальнейшего составления рейтинга исполнителей.

После формирования заказа исполнителю приходит отчет на почту, где указывается список его заказов с разделением на направления, специализацию и именем подобранного исполнителя.

Заключение

В качестве итога работы был разработан сайт для поиска удаленной работы. Сайт обладает удобным интерфейсом и быстрым подбором кандидата на задачу для исполнителя, а также удобным подбором вакансий для исполнителя. Данный сайт не подразумевает дополнительных финансовых вложений.

Таким образом все поставленные задачи были выполнены, но сайт может быть подвергнут модернизации, например, добавлением дополнительных параметров для подбора кандидата.

Список литературы

1. Django documentation [Электронный ресурс] – Режим доступа: <https://docs.djangoproject.com/>
2. ReportLab Tutorials [Электронный ресурс] – Режим доступа: <https://www.reportlab.com/>
3. Django framework [Электронный ресурс] – Режим доступа: <https://www.djangoproject.com/>
4. Section [Электронный ресурс] – Режим доступа: <https://www.section.io/>

DEVELOPMENT OF A WEBSITE FOR REMOTE JOB SEARCH

Marinova V.V. (valeria.marinova.00@gmail.com)
Ulyanovsk State Technical University, Ulyanovsk

This article describes the implementation of a site for remote job search with automatic selection of a candidate for a vacancy. The candidate is selected based on his rating, work experience and specialization.

Keywords: recommendation, dataset, specialization, selection, data, algorithm.

УДК 004.415

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ «WEB-ПРИЛОЖЕНИЕ ДЛЯ ТОРГОВЛИ ЛИМИТИРОВАННЫМИ ТОВАРАМИ»

Охотников Д.В. (den.ohotnikov@gmail.com)

Гуськов Г.Ю. (guskovgleb@gmail.com)

Ульяновский Государственный Технический
Университет, Ульяновск

В работе представлено описание разработанной информационной системы «Web-приложение для торговли лимитированными товарами». Система предназначена для автоматизации процессов покупки и продажи товаров ограниченной серии от артистов и брендов. Платформа агрегирует предложения и предлагает приобрести товар на единой площадке, подобный подход удобнее для покупателей, чем множество индивидуальных ресурсов.

Ключевые слова: товар, покупка, продажа, артист, лимитированная продукция, Express, React

Введение

Многие бренды и артисты заинтересованы в том, чтобы монетизировать свою популярность путем продажи лимитированных брендовых товаров. К брендовым товарам можно отнести профильные материалы, предметы гардероба, стандартные сувениры и прочие товары, которые несут в себе практическую пользу и в то же время уникальны по своему оформлению.

Подобная продукция обычно реализуется на специальных тематических мероприятиях. В данный момент, способ продажи не является оптимальным. Поскольку товар реализуется только на таких мероприятиях, большинство потенциальных покупателей не имеет возможности приобрести товар из-за их территориальных ограничений.

Требуется разработать информационную систему, представляющую из себя web-приложение для продажи продукции, которое позволит объединить бренды и артистов, предоставив возможность просмотра и покупки лимитированных товаров для пользователей.

Основные бизнес-процессы платформы:

- Размещение лимитированных товаров на площадке;
- Покупка и доставка товара пользователям;
- Учёт совершенных заказов.

Функциональные требования, предъявляемые к системе:

- Регистрация и авторизация в системе (Авторизация и регистрация проходит по адресу электронной почты, пользователь получает временный токен доступа с правами (обычный пользователь, продавец, администратор));
- Управление товарами (продавец имеет возможность создавать новый товар, предварительно указав производителя и категорию);
- Каталог товаров (все пользователи системы, вне зависимости от роли, имеют возможность просмотреть каталог товаров и подробную информацию о нем);
- Управление покупками (авторизованные пользователи имеют возможность добавлять и редактировать товары в корзине, совершать заказы и просматривать совершенные заказы);
- Доставка товаров пользователям (при оформлении заказа, пользователь выбирает способ доставки)
- Просмотр списка заказов (администратор и продавец могут просматривать заказы, совершенные пользователями системы).

Проектирование системы

Платформа представляет собой клиент-серверное web-приложение состоящее из компонентов:

- Бэкенд-приложение написано при помощи платформы Node.js (платформа, основанная на языке программирования JavaScript), фреймворк Express.js;
- Фронтенд часть – при помощи JavaScript-библиотеки React;
- Хранилище данных используется база данных PostgreSQL;
- Клиент взаимодействует с сервером при помощи библиотеки axios путём обращения к методам его API.

Веб-приложение разрабатывалось на основе Model-View-Controller.

В результате анализа ключевых прецедентов системы было выявлено 10 вариантов использования и 4 актора (неавторизованный пользователь, покупатель, продавец и администратор).

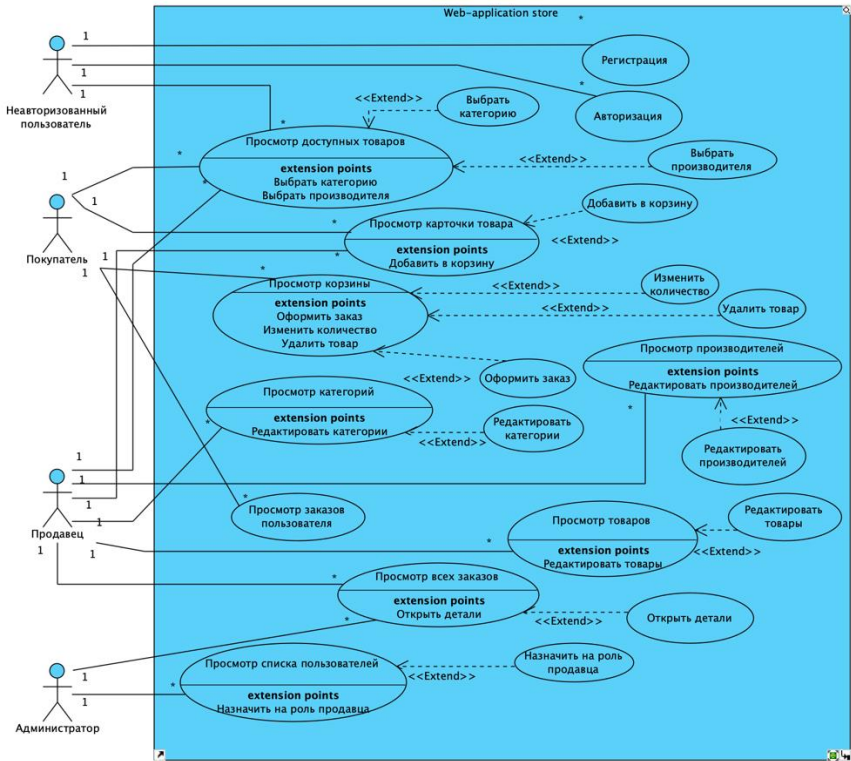


Рисунок 1 – Диаграмма вариантов использования

Разработанные приложения должны быть развёрнуты на локальном веб-сервере или хостинге, доступном для обращения по HTTP-протоколу.

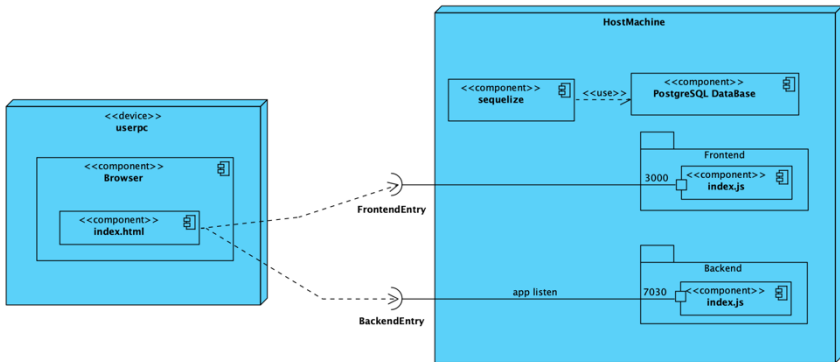


Рисунок 2 – Диаграмма развёртывания системы

Программная реализация системы

На странице каталога товаров показаны доступные карточки товаров, добавленные администратором, они содержат фотографию и описание товара, также слева от списка доступна сортировка товаров по категории и производителю (рисунок 3).

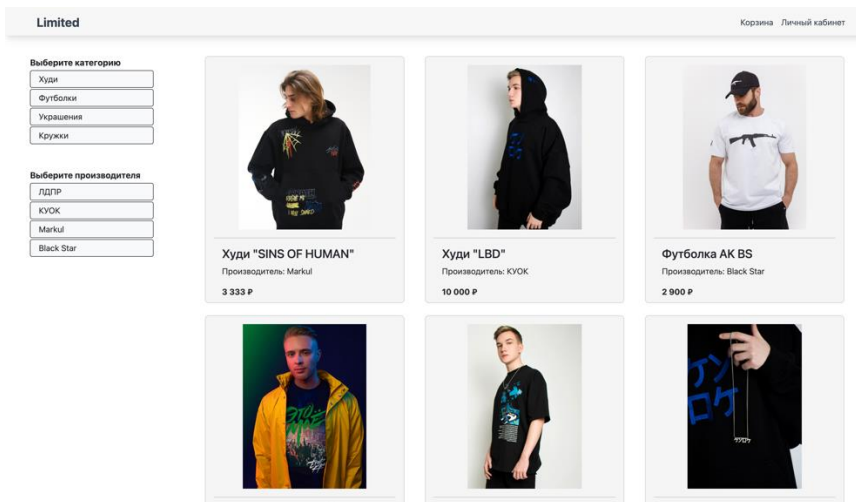


Рисунок 3 – Каталог товаров

Продавец может создавать карточки товаров для каталога товаров, а также редактировать их, изменяя название, фотографию, описание, стоимость и характеристики товара, в этом окне также доступно пополнение определенного товара на площадке.

После регистрации пользователю становится доступна работа с корзиной покупателя и оформления заказа. Создание заказа подразумевает заполнение персональной информации, указанной при регистрации пользователя в системе, после указания этих данных и выбора способа доставки заказ создается в системе. (рисунок 4).

Order form details:

- Name: Иван Иванов ✓
- Email: admin@mail.ru ✓
- Phone: +7 (999) 193-45-99 ✓
- Address: г. Ульяновск, ул. Ленина д. 55 ✓
- Check: проверка
- Button: Отправить

Рисунок 4 – Форма оформления заказа

Информацию по заказу можно получить на специальной странице, перейдя через личный кабинет (рисунок 5).

Order details:

- Дата заказа: 15.5.2022 16:0
- Статус заказа: Новый
- Имя, фамилия: Иван Иванов
- Адрес почты: admin@mail.ru
- Номер телефона: +7 (999) 193-45-99
- Адрес доставки: г. Ульяновск, ул. Ленина д. 55
- Комментарий: проверка

Название	Цена	Кол-во	Сумма
Футболка #ЭТОМОЕ Егор Крид	3000	3	9000
Итого			9000

Рисунок 5 – Заказ пользователя

Заключение

Результатом работы является полностью работоспособное клиент-серверное приложение «web-приложение для торговли лимитированными товарами», отвечающее поставленным требованиям.

Разработанное приложение позволит объединить артистов и бренды, повысит их узнаваемость, а также сократит время, затрачиваемое менеджерами на реализацию лимитированной продукции. Помимо этого, дистанционный формат продажи продукции окажет положительное влияние и на покупателей, позволяя получить подробную информацию о интересующем продукте.

В будущем рекомендуется расширение возможностей приложения. Например, проведение интернет-аукционов и интеграция системы проведения прямых эфиров с артистами и представителями брендов.

Список литературы

1. Корунова Н.В. Проектирование программного обеспечения: применение на практике / практикум по дисциплине «Проектирование информационных систем» / Н. В. Корунова. – Ульяновск: УлГТУ, 2019. – 69 с. - URL: <http://is.ulstu.ru/disc/pris>. – Режим доступа: для авториз. пользователей.
2. React – JavaScript-библиотека [Электронный ресурс] – URL: <https://ru.reactjs.org/docs/getting-started.html>.
3. Express - фреймворк веб-приложений Node.js [Электронный ресурс] – URL: <https://expressjs.com/ru/>.

DEVELOPMENT OF INFORMATION SYSTEM «WEB-APPLICATION FOR TRADE IN LIMITED EDITION GOODS»

Okhotnikov D.V. (den.ohotnikov@gmail.com)

Guskov G.Yu. (guskovgleb@gmail.com)

Ulyanovsk state technical University, Ulyanovsk

The paper presents a description of the developed information system "Web-application for trade in limited edition goods". The system is designed to automate the processes of buying and selling limited edition goods from artists and brands. The platform aggregates offers and offers to purchase goods on a single site, such an approach is more convenient for buyers than a lot of individual resources.

Keywords: goods, purchase, sale, artist, limited edition products, Express, React

РАЗРАБОТКА ВИДЖЕТА ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССОВ МАЛОГО БИЗНЕСА В СФЕРЕ ПРОДАЖ И ОКАЗАНИЯ УСЛУГ. СЛОЙ ИНТЕГРАЦИИ

Пиков К.А. (pikov_kirya@mail.ru)
Ульяновский Государственный Технический
Университет, Ульяновск

Статья посвящена реализации слоя интеграции с целью упрощения, уменьшения трудозатрат и ускорение процесса интегрирования систем. Разрабатываемый слой интеграции предназначен для оркестровки с некоторыми возможностями корпоративной сервисной шины. Разработанная система предлагает следующий процесс интеграции: определение интерфейсов систем, написание и загрузка, во время выполнения приложения, конфигурации, в соответствии с интерфейсом системы, тестирование, перенос на продуктивный сервер написанной конфигурации. В случае неудачи тестирования – изменение и загрузка конфигурации заново. Поставленные цели достигаются за счет отсутствия необходимости написания кода и последующих развертываний приложения в различных окружениях.

Ключевые слова: интеграция, оркестратор, обмен сообщениями, трансформация, коннектор, корпоративная сервисная шина.

Введение

В рамках исследования необходимо разработать систему – слой интеграции, целью разработки которой является упрощение, уменьшение трудозатрат и ускорение процесса настройки взаимодействий с сервисами, необходимость которых возникает в задачах оркестровки. Для достижения поставленной цели необходимо решить следующие задачи:

1. Изучить понятие интеграции, что под ней подразумевается и какие виды существуют.
2. Проанализировать программные аналоги, описать процессы интеграции КАК-ЕСТЬ и КАК-ДОЛЖНО-БЫТЬ.
3. Построить модель слоя интеграции.
4. Реализовать слой интеграции.

Объектом данного исследования являются процессы интеграции систем. Предметом исследования является проект и разработка системы, способной во время выполнения приложения настраивать компоненты, реализующие взаимодействие, посредством интерпретации конфигурации.

Анализ существующих решений

В качестве существующего решения предлагается рассмотреть платформу для автоматизации рабочих процессов и принятия решений Camunda.

Для платформы представлено расширение – коннекторы, которые представляют собой компоненты, поддающиеся настройке из, например, загружаемой диаграммы BPMN.

На момент написания статьи в этой библиотеке есть только 2 компонента, реализующие взаимодействие: HTTP и SOAP. Несмотря на скромный ассортимент готовых коннекторов, есть возможность написать индивидуальный – библиотека расширяема.

Однако, на практике, зачастую недостаточно только настроить коннекторы, появляется необходимость в мониторинге происходящего, в частности – компонентов, реализующих интеграцию, чего, на первый взгляд, пока нет в Camunda. Возможно, просмотр таких метрик, если и есть, то доступен только в платной версии панели администратора.

Еще, к недостаткам данного аналога можно отнести то, что описание коннекторов производится непосредственно в артефакте, к примеру, BPMN диаграмме. С одной стороны, можно рассматривать это как плюс – весь контекст в едином месте, но с другой стороны, на практике, возникает необходимость перенастроить коннекторы при переносе артефакта, к примеру, BPMN, на другое окружение - речь о точках доступа для коннекторов, тем самым придется либо менять параметры артефакта, либо изобретать индивидуальные решения. Таким образом, бизнес-логика в артефакте смешана с интеграционной составляющей.

Предлагаемое решение

Предлагаемое решение базируется на платформе Camunda. Эта платформа была выбрана потому что имеет открытый исходный код и развитое сообщество, хорошо документирована, некоторые решения доступны из коробки, имеет возможность горизонтального масштабирования, а также из-за интересов автора в расширении кругозора.

Создавать подобные системы с чистого листа достаточно тяжелое и

затратное дело, к тому же появляются ненужные риски. Поэтому для настройки интеграции с системами был выбран фреймворк Apache Camel, который упрощает настройку взаимодействия с различными системами. Apache Camel обеспечивает реализацию шаблонов интеграции корпоративных приложений, имеет широкую библиотеку готовых компонентов и, также как Camunda, открытый исходный код с богатой документацией и развитое сообщество.

Решение похоже на существующий аналог – коннекторы Camunda и также является расширяемым. Однако, отличие в способе конфигурации. Предлагаемое решение хранит конфигурацию отдельно от артефактов Camunda, из которых можно ссылаться на необходимый компонент по идентификатору.

Можно выделить несколько сущностей, которые позволяют описать интерфейс для взаимодействия:

1. Host – хранит описание сервера и параметров доступа к нему.
2. Connector – хранит описание интерфейса для взаимодействия, ссылается на Host по идентификатору.
3. Operation – сборное описание, может быть разных типов, хранит в себе описание интерфейса – Connector.

Выделенных сущностей достаточно, чтобы конфигурировать различные компоненты для взаимодействия с системами, так, например, можно объявить в приложении оркестраторе новую конечную точку, синхронный обмен сообщениями по http или sql запрос на выборку или вставку данных какой-либо БД.

Для примера, конфигурация хоста может выглядеть следующим образом:

```
{
  "id": "gateway",
  "host": "any.domain:50151",
  "auth": {
    "type": "basic",
    "login": "LH233EAE",
    "password": "WcS99QGjRabnzZV3o8"
  }
}
```

Листинг 1 – Пример конфигурации хоста

Конфигурация операции может выглядеть так:

```
{
  "id": "create_an_order",
  "type": "simple_message_exchange",
  "short_description": "Выполнит http POST запрос на хост gateway для
```

создания заказа, описана структура тела запроса, параметры подставляются из контекста процесса Samunda.",

```
"tracing_enabled": true,
"connector": {
  "type": "http",
  "host_id": "gateway",
  "method": "POST",
  "url": "/v1/order",
  "request": {
    "body": {
      "has_structure": "object",
      "contains_elements": {
        "goods": {
          "from_process_variable": "basket.items",
          "has_structure": "array",
          "contains_elements": {
            "goods_id": {
              "from_variable": "itemId"
            },
            "goods_name": {
              "from_variable": "itemName"
            },
            "retailer": {
              "from_variable": "retailer",
              "has_structure": "object",
              "contains_elements": {
                "retailerName": {
                  "from_variable": "name"
                },
                "retailerCode": {
                  "from_variable": "code"
                }
              }
            }
          }
        }
      }
    }
  }
}
```

Листинг 2 – Пример конфигурации хоста

Загрузив такую конфигурацию в приложение, операция на создание заказа будет доступна из артефакта Samunda – бизнес-процесса проведения заказа через компонент Сервисная задача, необходимо только ввести выражение:

```
operations.execute("create_an_order")
```

Листинг 3 – Пример выражения

Если возникнет необходимость поменять какие-либо значения, то прямо во время выполнения приложения можно загрузить новую конфигурацию и изменения вступят в силу без перезагрузки приложения. Также, Camunda предоставляет API для загрузки новых версий артефактов – BPMN диаграмм. Таким образом, разработка может вестись прямо на тестовом окружении, где представлены все системы, с которыми выполняется взаимодействие.

Стоит отметить, что при загрузке артефакта Camunda с описанием бизнес-процесса, использующего операции, проводится валидация файла на существование всех операций с такими идентификаторами.

Также предлагаемое решение из коробки поддерживает мониторинг созданных интеграционных компонентов посредством библиотеки для Camel и веб-консоли Hawtio, которая содержит различную очень полезную функциональность.

Немаловажным является наличие трассировки интеграционных компонентов, которая была разработана в рамках реализации слоя интеграции. При необходимости для любой операции можно включить подробное журналирование действий и разобрать, что происходит в системе в момент выполнения операции.

Планы на будущее, перспективы

Реализованная система является достаточной для использования, ее можно расширять и добавлять новую функциональность, к примеру, имеет место быть:

1. Разработка UI для настройки взаимодействий (против написания файлов-конфигураций вручную).
2. Реализация хранения конфигурационных файлов в постоянном хранилище (текущая реализация предполагает хранение конфигурационных данных в памяти приложения).
3. Добавление новых коннекторов.
4. Добавление сложной трансформации сообщений.
5. Вынести функциональность в отдельный jar, загрузить в публичное хранилище для последующего использования всеми желающими.
6. Разработка более сложной валидации конфигурационных файлов на согласованность с артефактами для Camunda.

Заключение

В ходе работы был спроектирован и реализован слой интеграции, позволяющий вести разработку и тестирование взаимодействий с системами непосредственно во время выполнения приложения, изменение конфигурации не требует перезапуска или развертывания новой версии приложения.

Для переноса в продуктивный сервер достаточно загрузить новую версию артефакта для Camunda – BPMN диаграмму, если она поменялась, и перенести конфигурацию операций, которая остается одинаковой для всех окружений, для разных сред меняется только конфигурация хостов.

Таким образом, поставленные задачи были выполнены, следовательно, цель достигнута.

Благодарности. Автор выражает благодарность всем, кто прочитал статью, и готов обсудить все вопросы и предложения по почте.

Список литературы

1. Документация Camunda [Электронный ресурс] – Режим доступа: <https://docs.camunda.org/manual/7.17/>
2. Документация Apache Camel [Электронный ресурс] – Режим доступа: <https://camel.apache.org/>
3. Форум Camunda [Электронный ресурс] – Режим доступа: <https://forum.camunda.io/>
4. Глоссарий процессного управления [Электронный ресурс] – Режим доступа: <https://piter-soft.ru/knowledge/glossary/process/>
5. Оркестровка и хореография: подходы к описанию композитных бизнес-процессов [Электронный ресурс] – Режим доступа: http://artamonoviv.ru/articles/business_process_orchestration_and_choreography/

DEVELOPMENT OF A WIDGET FOR AUTOMATING SMALL BUSINESS PROCESSES IN THE FIELD OF SALES AND SERVICES. INTEGRATION LAYER

Pikov K.A. (pikov_kirya@mail.ru)
Ulyanovsk State Technical University, Ulyanovsk

The article is devoted to the implementation of the integration layer in order to simplify, reduce labor costs and accelerate the process of system integration. The integration layer being developed is designed for orchestration with some of the capabilities of the enterprise service bus. The developed system offers the following integration process: defining system interfaces, code writing and loading, during application execution, configurations, in accordance with the system interface, testing, transferring the written configuration to a productive server. In case of testing failure – change and download the configuration again. The set goals are achieved due to the absence of the need to write code and subsequent deployment of the application in various environments.

Keywords: integration, orchestrator, messaging, transformation, connector, enterprise service bus.

УДК 004.42

РАЗРАБОТКА СИСТЕМЫ РЕГИСТРАЦИИ ДЛЯ РОЛЕВЫХ ОНЛАЙН-ИГР НА ВЕБ-ФРЕЙМВОРКЕ LARAVEL С ИНТЕГРАЦИЕЙ В DISCORD И MEDIAWIKI

Смирнов А.А.

Ульяновский государственный технический университет, Ульяновск

В работе рассматривается специфика разработки системы регистрации для ролевых онлайн игр с использованием средств веб-фреймворка Laravel и интеграцией в мессенджер Discord и вики на движке MediaWiki. Рассматривается экономическая эффективность от внедрения системы и срок окупаемости в расчёте на временные затраты.

Ключевые слова: сайт, веб-разработка, Laravel, Discord, MediaWiki

Введение

В настоящее время рынок изобилует крупными коммерческими игровыми проектами, что имеют достаточные средства для создания необходимой инфраструктуры и любых вспомогательных систем. Именно финансы, сопряженные риски и недостаточный опыт, который невозможно заработать, не начав, являются основными ограничителями выхода на рынок новых игр и проектов.

Использование аналогичных бесплатных или «ручных» решений, необходимых для поддержки игры, не всегда отвечает всем требованиям, пагубно влияет на качество конечного продукта, не позволяет выйти на новые рынки. Ещё хуже ситуация обстоит с некоммерческими проектами, которые не несут за собой цели получения прибыли, создаются мелкими группами разработчиков для небольших сообществ по интересам и не имеют достаточных средств.

В связи с этим возникает необходимость в создании дешёвого «промежуточного» решения, отвечающая большинству наиболее распространённых требований игровых проектов ролевой направленности.

Актуальность работы обусловлена наличием реально существующего игрового проекта с совокупным сообществом в несколько сотен игроков, для которого необходимо подобное решение. Система имеет открытый исходный код и может потенциально использоваться другими проектами, совокупная аудитория которых в русскоязычном сегменте достигает порядка нескольких тысяч человек.

На данный момент регистрация и все процессы, связанные с игрой, осуществляются посредством ручного редактирования вики на движке MediaWiki и средств Discord с использованием нескольких ботов. Это затрачивает много времени всех участников процесса (игроков и администрации), не позволяет оперативно обновлять и поддерживать актуальную информацию во всех частях системы.

В сравнении с текущей системой, новое решение позволит автоматизировать процесс, сэкономить время и избежать типичных ошибок.

Цели разработки

- Оптимизация процессов как для пользователей, так и для регистраторов и администрации
- Синхронизация данных при изменениях игровой системы
- Возможность расширения системы, интеграция её с другими системами (например, перенос навыков прямо в игру)
- Расширение аудитории – проведение работы над UX, привлечение новых пользователей за счет предоставления удобных средств взаимодействия с системой.

Описание разрабатываемой системы

Система представляет собой веб-сервис на платформе веб-фреймворка Laravel, интегрированный с мессенджером Discord и вики-движком MediaWiki.

Веб-сервис представляет собой сайт с возможностями регистрации любым сторонним пользователем, авторизации под логином и паролем, CRUD собственных персонажей, просмотра персонажей других пользователей.

Веб-сервис предполагает наличие четырёх ролей: игрока (роль по умолчанию), регистратора, гейм-мастера и администратора.

Игроки создают персонажей, которые проверяются регистраторами, прежде чем игрок сможет начать игру. После начала игры на зарегистрированном персонаже последующие изменения производятся гейм-мастерами. Администратор занимается управлением ролями и разрешениями пользователей, а также конфигурированием системы и заполнением необходимых для выбора при создании персонажа элементов справочников.

Интеграция с Discord представляет собой привязку аккаунта веб-сервиса к аккаунту Discord, отправку оповещений и автоматическое создание тикетов при выполнении ключевых операций.

Интеграция с MediaWiki представляет собой использование технологии OAuth 2.0 для авторизации на вики с использованием аккаунта веб-сервиса, а также вставку содержания справочников на страницы.

Назначение системы

Система предназначена для автоматизации процесса регистрации и управления персонажами пользователей, обеспечения синхронизации и непротиворечивости данных, предоставления адекватных и отвечающих современным требованиям UX средств взаимодействия с системой.

3 Экономическое обоснование

Расчет затрат на разработку программного продукта.

Этапы разработки:

1. Проектирование
2. Разработка
3. Внедрение

Каждый этап представлен набором задач для выполнения. Имеется резерв времени для тестирования (10%) и баг-фикса (10%).

Трудоёмкость для основного этапа (разработки) рассчитан с помощью метода функциональных точек (см. Приложение, Метод функциональных точек), так как он представляет наиболее адекватную оценку в 13,68 ч. в неделю.

Помимо этого, использовались методики оценки Use Case Points (10 ч. в неделю при рекомендуемых параметрах и 28 ч. при максимальных) и СОСОМО II (4,6 ч. в неделю).

Метод функциональных точек отражает максимально приближенную к реальности оценку трудоёмкости, учитывает сложность каждого функционального элемента и общих требований к проекту.

Все этапы разработки проводились одним человеком (разработчиком) с использованием собственного оборудования, помещения и интернета.

Таблица 1 – Трудоёмкость проектирования

Задача	Оценка, ч
Анализ требований, составление ТЗ	8
Проектирование, составление диаграмм	16
Тестирование	2,4
Баг-фикс, корректировка диаграмм	2,4
Итого	28,8

Таблица 2 – Трудоемкость разработки

Задача	Оценка, ч
CRUD персонажей	16
Процесс регистрации персонажей	26
Параметры персонажа, калькуляторы	28
Перки, CRUD, калькулятор	18
Администрирование, роли, разрешения, разграничение функционала	14
API для авторизации	2
Настройка OAuth 2.0 для MediaWiki	4
Настройка OAuth 2.0 для Discord	4
Интеграция с Discord – Оповещения, тикеты	12
Тестирование	12,4
Баг-фикс	12,4
Итого	148,8

Система внедряется на уже настроенный выделенный сервер, потому его настройка не учитывается в расчёте трудоемкости.

Таблица 3 – Трудоемкость внедрения

Задача	Оценка, ч
Установка и настройка инфраструктуры – веб-сервера, пакетов, зависимостей	3
Развертывание системы и связанных элементов (MediaWiki, бот Discord)	2
Конфигурирование систем	1
Настройка и создание БД, настройка резервного копирования	2
Итого	8

Так как проект является некоммерческим, не имеет финансирования, и разработчик не получал заработной платы, затраты представлены лишь во временном выражении – 185,6 ч.

Расчет экономических результатов от внедрения.

Внедрение автоматизированной системы регистрации позволяет сократить время регистрации каждого отдельного пользователя, время создания, проверки и регистрации персонажей.

Таблица 4 – Результаты от внедрения

№	Пользователь	Вид работ	До автоматизации, час	После автоматизации, час	Экономия времени, час	Повышение производительности труда, % (Pi)
1	Регистратор	Создание аккаунта пользователя	0,16	0,016	0,144	1000%
2	Игрок	Создание страницы персонажа	0,05	0,016	0,04	312,5%
3	Регистратор	Валидация числовых параметров персонажа	0,16	0	0,16	N/A
4	Регистратор	Регистрация аккаунта персонажа	0,03	0	0,03	N/A
5	Пользователь	Перенос справочных значений на страницу персонажа	0,16	0	0,16	N/A
6	Пользователь	Обновление страницы персонажа при изменениях справочных значений	0,06	0	0,06	N/A

Указанные в таблице выше работы проводятся по мере прибытия новых пользователей и при создании ими новых персонажей.

- В среднем на игровой проект в неделю приходит по 8 человек, 32 человека в месяц.
- Среднестатистический игрок создаёт нового персонажа раз в 3 месяца.

- Среднее количество активных игроков (с учетом приходящих и уходящих) составляет 100 человек.
- В среднем происходит около 3 правок системы в месяц, что вынуждает всех активных пользователей внести правки на страницы своих персонажей.

Данные показатели рассчитаны в среднем выражении на основе имеющихся данных за последние 2 месяца.

Исходя из указанных выше условий, рассчитаем экономию после внедрения системы для каждого вида операций:

- Регистрация новых пользователей
 - Включает «Создание аккаунта пользователя» и занимает 0,16 ч. После внедрения системы – 0,016 ч.
 - До внедрения – $32 * 0,16 \text{ ч.} = 5,12 \text{ ч.}$ в месяц
 - Экономия – $32 * 0,144 \text{ ч.} = 4,608 \text{ ч.}$ в месяц
- Регистрация нового персонажа
 - Включает «Создание страницы персонажа», «Валидация числовых параметров персонажа», «Регистрация аккаунта персонажа» и занимает 0,4 ч. После внедрения системы – 0,016 ч.
 - До внедрения – $(32 + 100 * 0,33) * 0,4 = 26 \text{ ч.}$ в месяц
 - Экономия – $(32 + 100 * 0,33) * 0,384 = 24,96 \text{ ч.}$ в месяц
 - 32 – для новых игроков и персонажей, $100 * 0,33$ – для активных пользователей и их новых персонажей раз в 3 месяца
- Внесение правок на страницы своих персонажей всеми пользователями
 - Составляет 0,06 ч. на 1 правку для 1 пользователя, 0 ч. после внедрения.
 - До внедрения – $0,06 * 3 * 100 = 18 \text{ ч.}$
 - Экономия – $0,06 * 3 * 100 = 18 \text{ ч.}$ для 3 правок в месяц и 100 активных пользователей

До внедрения затраты составляют 49,12 ч. в месяц, итоговая экономия – 47,568 ч. в месяц в расчёте на указанные выше показатели. Расходы в месяц составляют 1,552 ч.

Расчет сроков окупаемости проекта.

Для оценки эффективности внедрения разрабатываемой системы рассчитаем срок окупаемости. Так как проект некоммерческий, не имеет стабильного финансирования и расходов, которые стоит рассматривать в данном расчёте, он ведётся в затраченном времени на взаимодействие с системой.

Срок окупаемости составляет: $185,6 / (47,568 - 1,552) = 4,03$ месяца.

Таким образом, проект полностью окупится через 4 месяца эксплуатации. Можно сделать вывод об экономической эффективности системы регистрации персонажей в рамках отдельно взятого игрового проекта.

Заключение

В ходе выполнения работы была проведена оценка разрабатываемой в рамках выпускной квалификационной работы системы регистрации для ролевых онлайн-игр.

Была проведена оценка трудоемкости проекта тремя разными методами. Выбранным методом является метод функциональных точек, продемонстрировавший наиболее адекватную оценку с учетом всех необходимых условий.

Были рассчитаны затраты на проект. Так как проект является некоммерческим и не имеет стабильного источника финансирования, а вся работа выполнялась без оплаты, расчёт проведён в выражении затраченного времени.

В расчёте на затраченное время проект окупается через 4 месяца эксплуатации, из-за чего можно сделать вывод о его экономической эффективности.

Список литературы

1. [Laravel 2022] Laravel – Documentation [Электронный ресурс] // Техническая документация Laravel: [сайт]. URL: <https://laravel.com/docs/8.x> (дата обращения: 09.05.2022)
2. [Discord 2022] Discord – Developer Portal - Documentation [Электронный ресурс] // Техническая документация Discord API: [сайт]. URL: <https://discord.com/developers/docs/intro> (дата обращения: 09.05.2022)
3. [MediaWiki 2022] MediaWiki – База знаний [Электронный ресурс] // Техническая документация MediaWiki: [сайт]. URL: <https://www.mediawiki.org/wiki/MediaWiki> (дата обращения: 09.05.2022)
4. [Laravel 2022] Laravel – Documentation [Электронный ресурс] // Техническая документация Laravel: [сайт]. URL: <https://laravel.com/docs/8.x> (дата обращения: 09.05.2022)

DEVELOPMENT OF A REGISTRATION SYSTEM FOR ROLEPLAY ONLINE GAMES ON THE LARAVEL WEB- FRAMEWORK WITH INTEGRATION INTO DISCORD AND MEDIA WIKI

Smirnov A.A.

Ulyanovsk State Technical University, Ulyanovsk

The paper discusses the specifics of developing a registration system for online role-playing games using the Laravel web framework and integration into the Discord messenger and wiki on the MediaWiki engine. The economic efficiency from the implementation of the system and the payback period in terms of time costs are considered.

Keywords: website, web development, Laravel, Discord, MediaWiki

УДК 004.9

РАЗРАБОТКА МНОГОПОЛЬЗОВАТЕЛЬСКОЙ МОБИЛЬНОЙ ИГРЫ В ЖАНРЕ «СТРАТЕГИЯ»

Сяман С.В. (aksenova1648@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация многопользовательской мобильной игры в жанре “Стратегия”. Производится сравнение с аналогами в сфере мобильных игр.

Ключевые слова: мобильная игра, стратегия, игровой движок

Введение

В настоящее время мобильные игры становятся все более популярным явлением. Именно мобильные игры приносят большую часть денег в игровой индустрии. Также с наступлением пандемии и введением локдауна у людей появилась необходимость проводить свой досуг, не выходя из дома. И мобильные игры подходят здесь как-никак лучше, что подстегнул игровой мобильный рынок к дополнительному росту.

Современные мобильные игры переполнены всевозможными игровыми механиками, в которых сложно будет разобраться людям не знакомыми с играми. К тому же нынешняя игровая сессия может занимать достаточно долгое время, не позволяющее сыграть во время поездки в транспорте.

Таким образом было принято решение разработать многопользовательскую мобильную игру в жанре “Стратегия”. Данная игра призвана скоротать время пользователей не долгой игровой сессией. Игровой цикл разработан так, чтобы не занимать много времени.

Анализ существующих решений

Рассмотрим существующих на данный момент самых популярных представителей в жанре «Стратегия». Это Auto Chess War и Chaos Battle League.

Auto Chess War является представителем поджанра «Auto Chess». Этот жанр славится крайне сложными игровыми механиками и долгими

сессиями. Также стоит заметить, что «клонов» в подобном жанре бесчисленное множество и среди них довольно легко затеряться. Сама игра, кроме самого игрового процесса, предоставляет игрокам возможность прокачивать своих героев вне геймплея. Это можно сделать, как игровыми средствами, так и внеся в игру необходимое количество реальных денег, что может негативно сказаться на игровом процессе.

Таким образом Auto Chess War, сложна для понимания новым игрокам, имеет довольно долгую по продолжительности игровую сессию. Также в игре имеются «PayToWin» механики, что негативно сказывается на игровом процессе.

Chaos Battle League является копией другой популярной игры «Clash Royale». Как раз в честь этой игры был и назван поджанр. Суть этих игр заключается в том, что два игрока появляются на поле боя, где есть несколько дорог по которым игроки могут отправить своих персонажей, чтобы они разрушили главное здание противника. Также, как и в прошлой игре необходимо прокачивать своих персонажей, с которыми игрок играет, чтобы они стали сильнее. К тому же в игре присутствует рейтинговая система.

Проблема игр данного поджанра заключается в том, что прогрессия игрока и его игрового опыта сильно завязан на микротранзакциях, что плохо влияет на игровой процесс. Также игровая сессия может затянуться.

Предлагаемое решение

Таким образом было принято решение разработать мобильное приложение, в котором одна игровая сессия не будет занимать много времени, будет проста в освоении и не завязана на микротранзакциях.

Для реализации проекта было принято решение использовать игровой движок Unity3D, так как он предоставляет все необходимые инструменты для разработки мобильного приложения. Также был использован сервис Photon, который позволяет создавать лобби для игроков и обмениваться данными между игроками в одном лобби.

Целью данного проекта является многопользовательское мобильное приложение, которое позволяет проводить досуг не обращая внимание на длину игровой сессии, а также наслаждаться игровым процессом без «PayToWin» элементов, которые могут раздражать игроков.

Конечный продукт представляет собой многопользовательскую мобильную игру на платформу «Android» в жанре «Стратегия». Игровой процесс представляет собой противостояние двух игроков – каждый игрок по очереди ставит точки на ограниченное поле, и если две точки одного игрока стоят в одной горизонтальной или вертикальной линии и не перегорожены препятствиями, то проводится линия между ними и

засчитываются очки за каждую клетку, которая заняла линия. Если игрок пересечёт вражескую линию, то получит вдвое больше очков за каждую клетку, которую пересек. На поле боя также могут присутствовать различные элементы, позволяющие соединять точки различными способами. В функции игры входят: авторизация/регистрация при помощи Google-аккаунта, поддержание круглосуточного доступа к игровому процессу, составление таблицы рейтинга игроков, поиск игроков по своему рейтингу.

Заключение

В качестве итога работы была разработана многопользовательская мобильная игра. Игра обладает легким игровым процессом, что не будет отпугивать игроков, незнакомых с мобильными играми. Также игра не перегружена различными «PayToWin» элементами, что не будет раздражать игроков.

Таким образом все поставленные задачи были выполнены, но игра может быть подвергнут расширению, например, добавлением новых карт, различными кастомизациями игрового поля и добавлением различных инструментов, помогающих в бою с другими игроками.

Список литературы

1. Auto Chess War [Электронный ресурс] – Режим доступа: <https://play.google.com/store/apps/details?id=com.phoenix.autochess&gl=US>
2. Chaos Battle League [Электронный ресурс] – Режим доступа: <https://play.google.com/store/apps/details?id=com.thisgamestudio.chaosbattleleague&gl=US>
3. Unity3D [Электронный ресурс] – Режим доступа: <https://unity.com>
4. Photon [Электронный ресурс] – Режим доступа: <https://www.photonengine.com>
5. Тренды мобильных игр в 2021 году — обзор индустрии мобильных игр [Электронный ресурс] – Режим доступа: <https://vc.ru/flood/273356-trendy-mobilnyh-igr-v-2021-godu-obzor-industrii-mobilnyh-igr>

DEVELOPMENT OF A MULTIPLAYER MOBILE GAME IN THE «STRATEGY» GENRE

Syaman S.V. (aksenova1648@gmail.com)
Ulyanovsk state technical University, Ulyanovsk

This article describes the implementation of a multiplayer mobile game in the "Strategy" genre. A comparison is made with analogues in the field of mobile games.

Keywords: mobile game, strategy, game engine.

УДК 004.9

РАЗРАБОТКА СЕРВИСА РЕКОМЕНДАЦИЙ КОМПЬЮТЕРНЫХ ИГР НА ЯЗЫКЕ PYTHON

Тягин И.С. (tis3.1415@gmai.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация сервиса рекомендаций компьютерных игр. Так же приводится сравнение с существующими аналогами.

Ключевые слова: рекомендательная система, веб-сервис.

Введение

Игровая индустрия все больше растет с каждым годом и становится из самых больших индустрий в сфере развлечений. На сегодняшний день в компьютерные игры играет около 3.1 млрд людей, т.е. 40% всех людей. В год в мире выходит тысячи игр от маленьких инди-проект до крупных проектов с бюджетом в миллионы долларов. И исходя из такого количества игр перед пользователем встает вопрос выбора игры.

Для пользователя выбор игры может оказаться не самым простым, ведь ему нужно сначала изучить не маленький список игр, по возможности изучить отзывы на них, а затем основываясь на своих предпочтениях выбрать ту игру, которая, по его мнению, больше всего подходит.

С целью помощи пользователям в выборе подходящей им компьютерной игры на основе данных их профиля сервиса Steam было принято решение разработать сервис рекомендаций компьютерных игр. Данный сервис должен обеспечить обработку данных о пользователе и на основе этого предоставить пользователю набор предлагаемых пользователю компьютерных игр.

Анализ существующих решений

Steam – крупнейший в мире онлайн-сервис цифрового распространения игр и программ. Данная платформа распоряжается данными о свыше 50 тысяч игр и более чем 120 миллионах пользователей [1].

Сервис Steam предоставляет пользователям огромное количество функций, включая и рекомендательную систему, которая призвана помочь пользователю с подбором подходящих ему игр. Однако, данная функция подбора игр в большей степени преследует рекламную цель, чем помощь в подборе игр для пользователя.

Рекомендательная система Steam работает таким образом, что предлагаемые ею игры берутся из списка самых популярных игр среди все количества пользователей данной платформы, либо же предлагаются малоизвестные игры на основе жанров и меток игр, которые находятся в библиотеке пользователя. Этот принцип работы зачастую предлагает варианты, которые не подходят конкретному пользователю [2].

Другие аналогов подобных систем работают по сходим принципам, и при этом их не так много, как рекомендательных систем в других сфера, таких как музыка, кино и т.д.

Предлагаемое решение

Таким образом было принято решение о разработке рекомендательной системы компьютерных игр в виде веб-сервиса.

Для реализации алгоритма рекомендаций был выбран язык программирования Python, так как обладает большим количеством библиотек, предназначенных для работ с данными. Для написания веб-сервиса использовался Django – популярный и полнофункциональный серверный веб-фреймворк.

За ресурсы данных о играх и пользователях были взяты сервисы Steam и SteamSpy, так как являются самыми крупными источниками необходимой информации и обладают открытыми API, что упрощает доступ к этой информации.

Конечная система представляет собой веб-сервис, который на основе предоставленного пользователем аккаунта Steam определяет его предпочтения, затем ищет похожих пользователей с такими же предпочтениями, на основе которых выбираются игры, которые теоретически могут понравиться конкретному пользователю. Для полученных игр вычисляется прогноз на то, понравится ли конкретная игра пользователю, и из списка игр с успешным прогнозом делается рекомендация для пользователя.

Заключение

В качестве итога был разработан веб-сервис, предоставляющий пользователю функционал для получения рекомендаций. Данный сервис позволит пользователю сократить время, затрачиваемое на подбор игр

вручную, и увеличить точность рекомендаций по сравнению с аналогами.

Таким образом, поставленные задачи были выполнены в полной мере, но при этом сервис в будущем может быть расширен как новым функционалом, так и доработкой существующего алгоритма.

Список литературы

1. Steam - Википедия [Электронный ресурс] – Режим доступа: <https://ru.wikipedia.org/wiki/Steam>
2. Рекомендательная система Steam [Электронный ресурс] – Режим доступа: <https://store.steampowered.com/explore/>

DEVELOPMENT OF A COMPUTER GAME RECOMMENDATION SERVICE IN PYTHON

Tyagin I.S. (tis3.1415@gmail.com)
Ulyanovsk State Technical University, Ulyanovsk

This article describes the implementation of the computer games recommendation service. A comparison with existing analogues is also given.

Keywords: recommender system, web service.

УДК 004.9

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ДЛЯ БУХГАЛТЕРСКОГО УЧЕТА В МАЛОМ БИЗНЕСЕ НА .NET FRAMEWORK

Утицких Д.А. (samprp458804@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация автоматизированной системы для бухгалтерского учета в малом бизнесе. Производится сравнение с аналогами в сфере бухгалтерского учёта.

Ключевые слова: бухгалтерия, автоматизация, бизнес, десктоп.

Введение

На данный момент, каждый мало-мальски мыслящий Индивидуальный Предприниматель желает увеличить эффективность и скорость работы своих сотрудников, но при этом не у каждого возникает желание разбираться с 1С, ведь обучение и интеграция занимает достаточно времени, при этом зачастую большинство функций необходимы здесь и сейчас, в простом и удобном формате.

Таким образом было принято решение разработать автоматизированную систему для бухгалтерского учёта в малом бизнесе. Основной задачей которого является автоматизация бухгалтерского учёта по упрощённой схеме. Система представляет собой десктопное приложение, который осуществляет автоматизацию бухгалтерского учёта по упрощённой схеме.

Анализ существующих решений

Самым популярным и самым продвинутым на рынке продуктом является 1С:Предприятие.

Данный продукт обладает огромным количеством функционала и гибкостью его настройки, что делает его лучшим среди конкурентов. Продукт предоставляет массу возможностей: от ведения складского учёта, инвентаризации, бухгалтерского учёта, кадрового учёта до составления отчётов, банковских операций и так далее.

С учётом множества тиражных версий (таких как 1С:Управление нашей фирмой, 1С:Упрощенка) представляющих собой надстройки над системой, функционал продукта увеличивается ещё больше, подстраиваясь под нужды каждого отдельного предпринимателя. Но в этом и кроется проблема, поскольку разных настроек много пользователь теряет в выборе настройки под себя.

Стоимость данного продукта от 3 до 300 тысяч рублей, в зависимости от поставки, но также предполагается и подписочная система, стоимость которой от 500 до 1200 рублей в месяц.

Самым продвинутым на данный момент аналогом 1С является ЕКАМ, созданная на онлайн-платформе и ориентированная на представителей малого и среднего бизнеса, для которых иметь дело с громоздким и дорогим софтом просто не выгодно.

ЕКАМ, в отличие от 1С имеет довольно простой интерфейс и предлагает большинство необходимых возможностей: ведение складского и товарного учёта, интеграцию с онлайн-кассой, интеллектуальную систему закупок, CRM модули, автоматизацию инвентаризации и так далее.

Стоимость данного продукта составляет от 960 рублей в месяц.

Предлагаемое решение

Таким образом было принято решение разработать доступную по стоимости и эксплуатации и обучению систему.

Для реализации проекта было принято решение использовать NetFramework поскольку дизайн приложения должен быть весьма простым для понимания.

Целью данного проекта является система, основанная на запросах к веб-серверу 1С и обработки бизнес-логики внутри приложения.

Конечная система представляет собой десктопное приложение с возможностью ведения бухгалтерского и складского учёта: ведение номенклатуры, остатков, розничной продажи. Система будет обрабатывать данные и отсылать уже готовые, чтобы снизить нагрузку на сервер.

Система будет разворачиваться на собственном аппаратном обеспечении. Цена вопроса от 500 рублей в месяц.

Заключение

В качестве итога работы был разработана автоматизированная система для бухгалтерского учёта в малом бизнесе. Система обладает легким и понятным интерфейсом, а также приемлемой скоростью работы, так как

обработка данных ведется на клиенте. Для работы с системой нет необходимости разбираться с огромным числом настроек, а стоимость намного меньше, чем у конкурентов.

Таким образом все поставленные задачи были выполнены, но система может быть подвергнута расширению, добавлению другого функционала 1С.

Список литературы

1. ТОП-10 аналогов программы «1С: Торговля и склад»: полезный обзор для розницы [Электронный ресурс] – Режим доступа: <https://www.ekam.ru/blogs/pos/analogi-programmy-1s-torgovlya-i-sklad>
2. Система программ 1С:Предприятие [Электронный ресурс] – Режим доступа: <https://v8.1c.ru/>
3. ЕКАМ [Электронный ресурс] – Режим доступа: <https://www.ekam.ru/>

DEVELOPMENT OF AN AUTOMATED SYSTEM FOR ACCOUNTING IN A SMALL BUSINESS ON THE .NET FRAMEWORK

Utitskikh D.A. (samprp458804@gmail.com)
Ulyanovsk state technical University, Ulyanovsk

This article describes the implementation of an automated system for accounting in a small business. A comparison is made with analogues in the field of accounting.

Keywords: accounting, automation, business, desktop.

УДК 004.02

ОБНАРУЖЕНИЕ ПРЕЗЕНТАЦИОННЫХ АТАК ПРИ РАСПОЗНАВАНИИ ЛИЦ НА ОСНОВЕ ДЕТЕКТИРОВАНИЯ ДВИЖЕНИЯ ВЕК

Фаворская М.Н. (favorskaya@sibsau.ru)

Кутузова М.О. (marina.kutuzova.00@inbox.ru)

Сибирский государственный университет науки и
технологий имени академика М. Ф. Решетнева,
Красноярск

В докладе рассматривается проблема обнаружения презентационных атак в системе распознавания лиц. Приведена классификация презентационных атак, а также подробно описан метод их обнаружения на основе детектирования движения век. В процессе работы был обучен детектор YOLOv5 с использованием собственного набора данных.

Ключевые слова: презентационные атаки, распознавание лиц, искусственные нейронные сети

Введение

В настоящее время во многих практических задачах необходимо выделять индивидуальные биометрические характеристики. Так, в системах пропускного контроля разного назначения, городского видеонаблюдения чаще всего используется метод считывания биометрии лица. Современные системы верификации и идентификации пользователей, построенные на основе методов глубокого обучения, способны с высокой точностью справляться с этой задачей. Однако при этом, начиная с 2017 года, возросло и количество атак на системы биометрического распознавания лиц. Таким образом, возникает актуальная задача по предотвращению атак, направленных на подмену биометрических признаков (так называемых спуфинг-атак).

Классификация спуфинг-атак

Технология распознавания лиц является одной из самых предпочтительных благодаря бесконтактности и минимальным

требованиям взаимодействия с пользователем, однако она уязвима к атакам разных видов. Спуфинг-атаки (spoofing attacks) представляют собой кибератаку, при которой человек или программа успешно маскируются под другого пользователя путем фальсификации данных с целью получения несанкционированного и незаконного преимущества. Наиболее популярными видами атак являются так называемые презентационные атаки (presentation attacks), которые подразделяются на атаки с использованием печати (print attack), атаки воспроизведения (replay attack) и атаки с использованием 3D масок (3D mask attack) [1]. Каждый вид презентационной атаки включает разные подвиды, одни из которых хорошо поддаются детектированию, но и имеются и такие атаки, обнаружение которых не представляется возможным ни одним устройством захвата (например, профессиональное гримирование с применением специальных масок).

В данном случае рассматриваются атаки печати, которые представляют собой распечатанные фотографии, изображения со смартфонов, планшетов и других электронных устройств. Решение о подлинности принимается путем анализа небольшого видеоролика на предмет движения век. Иными словами, «живость» лица оценивается по наличию или отсутствию моторики век.

Метод обнаружения презентационных атак

Принимая во внимание тот факт, что на сегодняшний день получить фотографию какого-нибудь человека не является трудной задачей, предъявление видеочкаме ре распечатанных фотографий, снимков с экрана телефона или непродолжительных видеороликов является легко осуществимым несанкционированным действием. Существует ряд методов для борьбы с презентационными атаками, которые разделяются на активные и пассивные методы. Активные методы запрашивают от пользователя выполнение какого-либо действия: повернуть голову, улыбнуться, моргнуть и т.д. Пассивные методы обычно используют одно входное изображение и анализируют текстуру области лица. Отметим, что возможно использование дополнительного оборудования (инфракрасные камеры, тепловизоры, 3D камеры) как эффективного способа предотвращения презентационных атак. Однако высокая стоимость специальных устройств захвата и сложность интеграции делают такие методы малоприменимыми на практике.

Одним из наиболее эффективных решений, относящийся к активным методам, состоит в определении «живости» лица, например, моргание глаз или движение губ. В данной работе в качестве основного метода обнаружения спуфинг-атаки выбран метод отслеживания движения век по

видеопоследовательности.

Для реализации модуля обнаружения «открытый-закрытый» глаз было решено использовать детектор YOLOv5 [2], который предсказывает координаты определенного количества ограничивающих прямоугольников (bounding boxes), согласует их с результатами классификации объекта и затем корректирует местоположение ограничивающих прямоугольников. Отметим, что версия детектора YOLOv5 является переписанной на языке Python версией детектора YOLOv4. Следует отметить, что модель детектора YOLOv5 была существенно модифицирована в плане скорости обнаружения объектов и возможности совмещения обнаружения и распознавания. Такая модель позволяет решать задачу детектирования движения век по видеопоследовательности в режиме реального времени.

В качестве обучающей выборки использовался собственный набор данных, состоящий из примеров, взятых из открытого набора данных [3] и изображений лиц с закрытыми глазами, созданных самостоятельно, с целью создания сбалансированного набора данных. Примеры изображений из собственного набора данных, собранные из социальных сетей и сети интернет, приведены на рисунке 1.



Рисунок 1 – Примеры изображений из собственного набора данных

В выборке присутствуют лица с открытыми и закрытыми глазами. Набор данных содержит около 1000 изображений размером 480×640 пикселей, из которых 70% изображений использовались для обучения модели, а 30% оставшихся изображений использовались в качестве тестовой выборки. Точность полученной модели составила 85,83% при обучении на 100 эпохах. Таким образом, с помощью обученной нейронной сети система может предсказывать стояние глаз в каждом кадре видеопоследовательности: закрыты или открыты, в случае, если система получает сценарий «открытые-закрытые-открытые» глаза,

следовательно, пользователь моргнул.

Для детектирования движения век необходимо определять расположение глаз человека на каждом кадре видеопоследовательности. В качестве метода распознавания лица и глаз человека были выбраны заранее обученные классификаторы Хаара [4]. Программный модуль реализован с использованием библиотеки OpenCV [5], которая содержит методы для работы с видеопоследовательностью, а также обнаружения лиц и глаз. Для точности распознавания глаз после детектирования лица, обрезанный кадр делится пополам и на каждой из частей определяется правый и левый глаз соответственно.

Для определения «живости» лица, полученные изображения глаз передаются в обученную модель нейронной сети для определения состояния глаза. Впоследствии, строится последовательность, где «0» обозначает то, что глаз открыт, и «1» в противном случае. Проанализировав полученную последовательность можно сделать вывод о «живости» изображения в видеопоследовательности.

Демонстрация входных данных и вывода программного модуля при обнаружении «живости» лица представлена на рисунке 2а. При обнаружении паттерна «открытый-закрытый-открытый» глаз, строится прямоугольник вокруг лица пользователя и в случае, если имеется набор данных для распознавания персоны, то выше указывается имя с постфиксом «real».

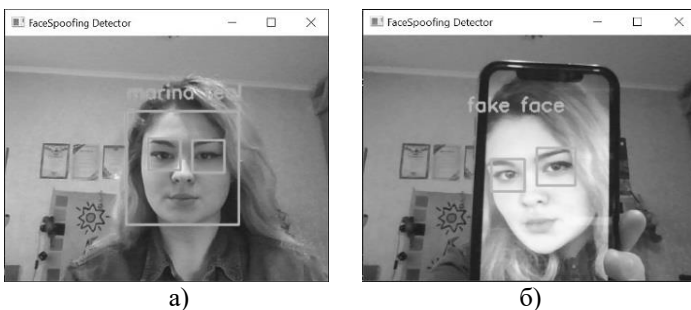


Рисунок 2 – Примеры распознавания: а) подлинное лицо, б) презентационная атака

При обнаружении презентационной атаки лицо на захваченном кадре прямоугольником не выделяется, а выше появляется надпись, которая сообщает, что в видеопоследовательности представлена фотография или другой вид спуфинг-атаки. Демонстрация вывода представлена на рисунке 2б.

Заключение

В докладе рассмотрен метод обнаружения презентационных атак на основе отслеживания движения век. Благодаря обученным классификаторам Хаара и детектору YOLOv5 разработан программный модуль, определяющий «живость» лица и работающий в реальном режиме времени. Для обучения детектора YOLOv5 был создан собственный набор данных лиц с открытыми и закрытыми глазами, что позволило достичь точности 85,83% обнаружения презентационных атак.

Однако следует отметить, что в реальных условиях возможны дополнительные манипуляции со стороны несанкционированных пользователей, например, моргание через вырезанные отверстия в фотографии, что требует усовершенствования алгоритмов определения «живости» лица.

Список литературы

1. Abdullakutty F., Elyan E., Johnston P. A review of state-of-the-art in face presentation attack detection: From early development to advanced deep learning and multi-modal fusion methods // Information Fusion. 2021, vol. 75. P. 55–69.
2. Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y.M.: YOLOv4: Optimal speed and accuracy of object detection. CoRR arXiv preprint, arXiv:2004.10934v1. 2020. P. 1–17.
3. Human Faces [Электронный ресурс] [сайт]. URL: <https://www.kaggle.com/datasets/ashwingupta3012/human-faces/code> (дата обращения: 11.03.2022).
4. Классификатор Хаара [Электронный ресурс] [сайт]. URL: <http://sibnigmi.ru/ait/vision> (дата обращения: 22.03.2022).
5. Документация OpenCV [Электронный ресурс] [сайт]. URL: <https://opencv.org/> (дата обращения: 22.03.2022).

DETECTION OF PRESENTATION ATTACKS IN FACE RECOGNITION BASED ON EYELID MOVEMENT

Favorskaya M.N. (favorskya@sibsau.ru)
Kutuzova M.O. (marina.kutuzova.00@inbox.ru)
Reshetnev Siberian State University of Science and
Technology, 31 Krasnoyarsky Rabochy ave., Krasnoyarsk,
660037 Russian Federation

The report deals with the problem of detection of presentation attacks in the face recognition system. A classification of presentation attacks is given, and a method for their detection based on the eyelid movement is described in detail. During experiments, the YOLOv5 detector was trained using our own dataset.

Keywords: presentation attacks, face recognition, artificial neural networks.

УДК 004.9

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ПО ПОДБОРУ КОМПЛЕКТУЮЩИХ ДЛЯ ПК

Хоменко Г.К. (goxaniguana@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В данной статье описывается реализация интеллектуальной системы по подбору комплектующих для ПК. Производится сравнение с аналогами в сфере конфигураторов персональных компьютеров.

Ключевые слова: персональный компьютер, подбор, критерии, данные, алгоритм, Django.

Введение

За последние два года огромное количество людей столкнулись с таким явлением как эпидемиологическая обстановка. В связи с этим людям пришлось работать, учиться и развлекаться из дома. И в первую очередь они столкнулись с банальной проблемой отсутствия домашнего компьютера. И начав поиски компьютеров для покупки сталкиваются с проблемами готовых рабочих станций от самых разных производителей.

В первую очередь с ценой, так как самому собрать компьютер из отдельных комплектующих гораздо дешевле чем купить готовый компьютер от именитого производителя. Но на рынке огромное количество производителей, стандартов и комплектующих, из-за чего, собрать хороший персональный компьютер, не имея знаний и опыта – задача невыполнимая.

Таким образом имея опыт в сборке, обслуживании и ремонте компьютеров и комплектующих, было принято решение о разработке онлайн сервиса, который поможет выбрать полностью совместимые и высококачественные комплектующие для самостоятельной сборки ПК.

Для разработки был выбран язык программирования Python в купе с библиотекой Django, для обеспечения бесперебойной работы сервиса и удобной работы с базой данных.

Анализ существующих решений

Рассмотрим существующие на данный момент программные решения данной проблемы.

Одна из крупнейших розничных сетей DNS предлагает конфигуратор ПК, который можно собрать только в их магазине. У данного решения для существует несколько очевидных проблем для конечного пользователя, в первую очередь это ограниченность одним магазином. Ассортимент может различаться в зависимости от города и времени сборки, а также такое решение не дает представления о комплектующих представленных в других магазинах. Если же рассматривать конфигуратор DNS более подробно, то можно заметить неправильный подбор комплектующих, причем любых, от кулера центрального процессора до блока питания. Сам же конфигуратор иногда предлагает заведомо несовместимые детали и уже на финальном этапе утверждения сборки сообщает о проблемах совместимости. Так же конфигуратор умалчивает о возможных проблемах, например, необходимости обновления BIOS, или крепления кулеров, которые к определённым сокетам продаются отдельно.

По итогу получаем несовершенное решение практически со всех сторон. Из плюсов стоит отметить возможность моментально заказать выбранные комплектующие в ближайший магазин или центр выдачи.

Из зарубежных аналогов стоит рассмотреть PCPartPicker. Хорошо зарекомендовавший себя сервис, существующий отдельно от магазинов. Данный сервис не представлен в России и не поддерживающий русский язык. В остальном же сервис представляет из себя достаточно посредственную систему подбора, которая повторяет ошибки сервиса от DNS, такие как: предлагает выбрать несовместимые комплектующие и только на этапе подтверждения сборки сообщает о несовместимости, предлагает чрезвычайно дорогие комплектующие к бюджетным сборкам и конечно же упускает незначительные моменты, как например наличие или отсутствие кулера в комплекте с процессором, что может очень сильно сказаться на цене финального компьютера.

Ну и конечно же эти решения лишены любых других методов подбора комплектующих кроме банальной проверки на совместимые стандарты между собой.

Предлагаемое решение

Изучив конкурентов присутствующих на рынке, принято решение о создании сервиса, который сможет подбирать комплектующие основываясь не только на проверке совместимых стандартов в сфере ПК, но и на экспертной оценке комплектующих.

Для реализации задуманного проекта выбран язык программирования Python с библиотекой Django. Так как это, очень мощный инструмент для создания сайтов-сервисов, на котором работают такие гиганты как: Instagram, The Washington Times и YouTube.

Итоговое решение представляет собой веб-сайт с возможностью выбрать комплектующие для будущего компьютера.

Попадая на главную страницу, пользователь может либо начать самостоятельную сборку, либо посмотреть уже готовые сборки введя ID сборки в специальное поле, таким же образом можно делится сборками с другими, достаточно просто передать ID, и другой человек сможет ознакомиться с финальной версией сборки компьютера.

Выбор комплектующих начинается с центрального процессора и сразу же можно заметить отличительную особенность сервиса, на выбор предлагается три списка комплектующих сортированных по экспертной оценке и результатам формул, например: «Лучшие производительные процессоры», «Лучшие по соотношению цена/качество», «Лучшие бюджетные процессоры». У каждой категории комплектующих индивидуальные характеристики, стандарты и возможности. Следовательно, у каждой категории разные критерии оценивания и формулы оценивания. Компьютер собирается в определённом порядке и выбранные комплектующие определяют требования к тем комплектующим которые только предстоит выбрать. Схема работы: сначала формируется список полностью совместимых комплектующих, которые потом делятся на три списка и сортируются в порядке убывания согласно результатам формул и экспертных оценок.

Таким образом пользователь получает не просто список полностью совместимых комплектующих, но список, каждая деталь в котором была оценена относительно рынка и получила подтверждение что она одна из лучших в представленной категории.

Заключение

В заключении работы был разработан интеллектуальный веб-сервис по подбору комплектующих для персональных компьютеров. Сервис предоставляет выбор комплектующих из категорий, которые закрывают практически все возможные требования среднестатистического пользователя, которые он может выдвинуть к персональному компьютеру. Количество ошибок совместимости между комплектующими снижено до возможного минимума, а заготовленные советы в процессе подбора комплектующих помогут пользователям собирать собственные компьютеры.

Все поставленные задачи были выполнены, но сервис легко поддаётся

расширению, благодаря продуманной структуре базы данных, а также, путём добавления новых критериев оценивания и формул для них.

Список литературы

1. PCPartsPicker [Электронный ресурс] – Режим доступа: <https://pcpartpicker.com/about/>
2. DNS Конфигуратор [Электронный ресурс] – Режим доступа: <https://www.dns-shop.ru/configurator/>
3. Django framework [Электронный ресурс] – Режим доступа: <https://www.djangoproject.com/>
4. Конструирование программного обеспечения. Учебное пособие. [Электронный ресурс] – Режим доступа <https://www.sibsau.ru/sveden/edufiles/144231/>
5. ГОСТ 34.201-89 – Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем. [Электронный ресурс] – Режим доступа: <http://www.gostedu.ru/11319.html/>

DEVELOPMENT OF MEDIA CONTENT CONCENTRATION SERVICE

Khomenko G.K. (goxaniguana@gmail.com)
Ulyanovsk state technical University, Ulyanovsk

This article describes the implementation of an intelligent system for the selection of components for a PC. A comparison is made with analogues in the field of personal computer configurators.

Keywords: personal computer, selection, criteria, data, algorithm, Django.

УДК 004.415

РАЗРАБОТКА ПРИЛОЖЕНИЯ «ИНФОРМАЦИОННЫЙ ПОРТАЛ ОБ ИНВЕСТИЦИЯХ»

Черноморченко А.В. (alinok2000@gmail.com)

Романов А.А. (romanov73@gmail.com)

Ульяновский Государственный Технический
Университет, Ульяновск

В работе представлено описание разработанного веб-приложения: «Информационный портал об инвестициях», предназначенного для упрощения задачи поиска и обработки информации, предоставления возможности обучения и быстрого ознакомления с актуальными руководствами, как для начинающих, так и для опытных инвесторов.

Ключевые слова: проектирование, информационные системы, разработка, программный продукт, инвестиции

Введение

В настоящее время инвестиционная сфера сформировалась как отдельный вид бизнеса и её популярность продолжает набирать обороты. С каждым годом все больше и больше людей хотят освоить процесс вложений своих средств и начать вести собственный бизнес, как начинающий инвестор. Современные информационные порталы перегружены различным контентом, разобраться в котором новичку крайне сложно.

Требуется разработать информационный портал с понятным интерфейсом для упрощения задачи поиска и обработки информации, предоставления возможности обучения и быстрого ознакомления с актуальными руководствами, как для начинающих, так и для опытных инвесторов.

Бизнес-требования заключаются в упрощении процесса сбора и обработки информации об инвестировании, а также визуализации собранной статистики для администратора портала.

Пользовательские требования:

- Возможность обучаться и получать актуальную информацию об

инвестировании на портале (для пользователей)

- Возможность взаимодействия между пользователями портала;
- Возможность просмотра статистики (для администратора);
- Возможность управления порталом (для администратора);
- Возможность регистрации и авторизации пользователей на портале.

Функциональные требования, предъявляемые к системе:

- Разграничение ролей (в веб-приложении должна присутствовать функциональность по разграничению ролей участников портала на пользователей и администратора);
- Аутентификация пользователей (в веб-приложении должна присутствовать функциональность для регистрации и авторизации пользователей портала);
- Управление личными данными (в веб-приложении должна присутствовать функциональность для изменения личных данных пользователя);
- Управление публикациями, категориями и тегами (в веб-приложении должна присутствовать функциональность, доступная администратору, позволяющая просматривать, создавать, редактировать и удалять публикации, категории и теги портала);
- Поиск и фильтрация публикаций (в веб-приложении должна присутствовать функциональность, позволяющая искать и фильтровать необходимую публикацию, по ключевым словам, категории или тегу);
- Комментирование публикаций (в веб-приложении должна присутствовать функциональность, доступная пользователям, позволяющая создать и удалять собственные комментарии на портале);
- Модерация комментариев (в веб-приложении должна присутствовать функциональность, доступная администратору, позволяющая отслеживать комментарии пользователей и, при необходимости, их удалять);
- Отслеживание пользователей (в веб-приложении должна присутствовать функциональность, доступная администратору, позволяющая отслеживать активность пользователей);
- Отслеживание статистики публикаций (в веб-приложении должна присутствовать функциональность, доступная администратору, позволяющая отслеживать статистику публикаций).

Проектирование системы

Портал был реализован в виде клиент-серверного приложения с использованием фреймворка Laravel (язык программирования – PHP), шаблонизатора Blade (JavaScript, html, css). В качестве СУБД была выбрана MySQL. Взаимодействие клиентской и серверной части происходит с помощью технологии RESI-API.

Для создания веб-приложения был применен архитектурный паттерн MVC (Model-View-Controller). Под понятием «MVC» подразумевается разделение системы на модель-представление-контроллер.

Для выявления вариантов использования системы была спроектирована соответствующая диаграмма. В итоге было выявлено 22 варианта использования (рисунки 1 и 2).



Рисунок 1 – Диаграмма вариантов использования, часть 1

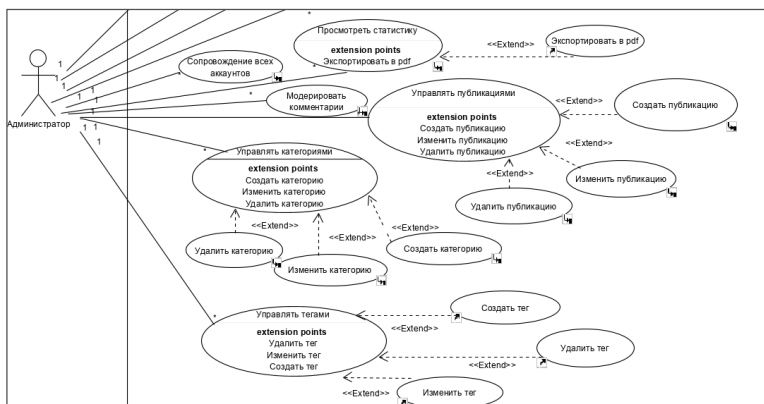


Рисунок 2 – Диаграмма вариантов использования, часть 2

Для детализации структуры разработанного ПО (программного

обеспечения) – представления основных компонентов системы, а также их взаимодействия, была спроектирована диаграмма компонентов (рисунок 3).

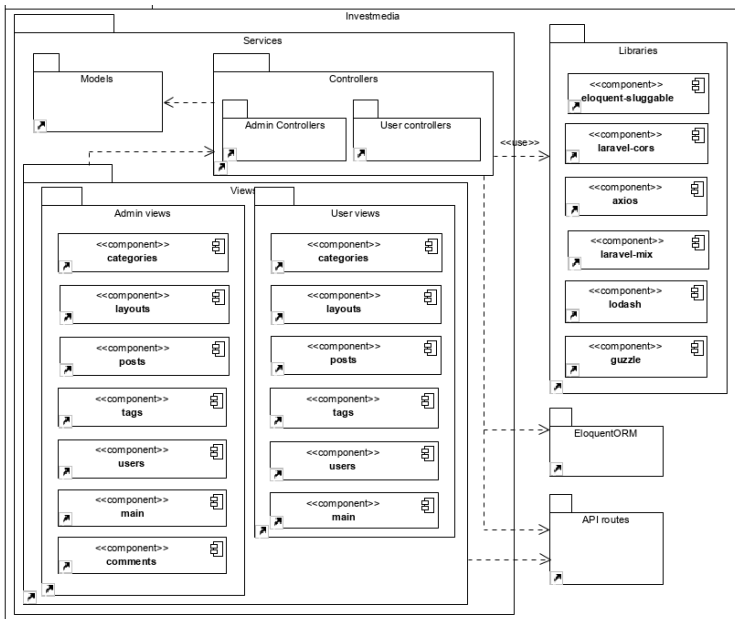


Рисунок 3 – Диаграмма компонентов системы

Программная реализация системы

На главной странице веб-приложения присутствует список всех публикаций портала с датой публикаций, категорий и количеством просмотров, для удобства просмотра информации была внедрена технология пагинации (разделение целого списка на отдельные пронумерованные страницы).

Кроме того, на данной странице присутствует баннер, рейтинг публикаций, список категорий, навигационная панель, предполагающая переход на другие страницы системы, путем нажатия на соответствующую кнопку, а также поле для поиска необходимой публикации по ключевому слову (рисунок 4).



Рисунок 4 – Главная страница приложения

При переходе на вкладку «Регистрация» пользователь попадает на страницу регистрации новых пользователей (рисунок 5), где присутствует форма с заполнением необходимых полей.

Если у пользователя уже есть аккаунт – он может перейти по соответствующей ссылке и попасть на страницу авторизации, аналогично присутствует ссылка для возврата на главную страницу.

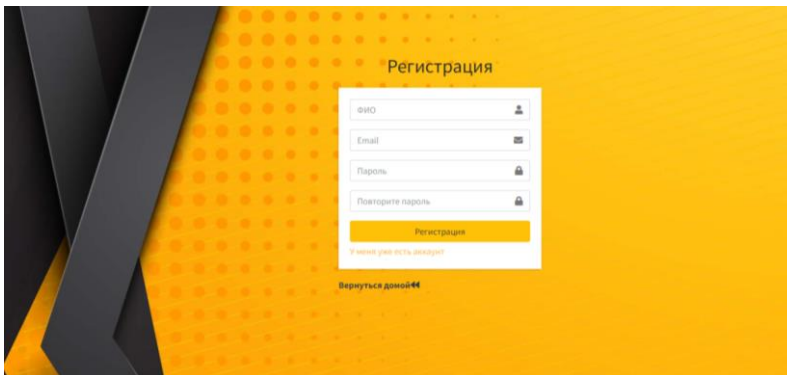


Рисунок 5 – Страница регистрации

После успешной регистрации или авторизации в системе у пользователя появляется возможность взаимодействовать с другими пользователями веб-приложения – авторизованные пользователи могут комментировать публикации портала, для этого пользователю необходимо открыть любую публикацию и заполнить поле для комментария внизу страницы.

Пользователь имеет возможность перейти в свой личный кабинет для управления комментариями и личными данными (рисунок 6). При нажатии кнопки «Удалить» выбранный комментарий удаляется.

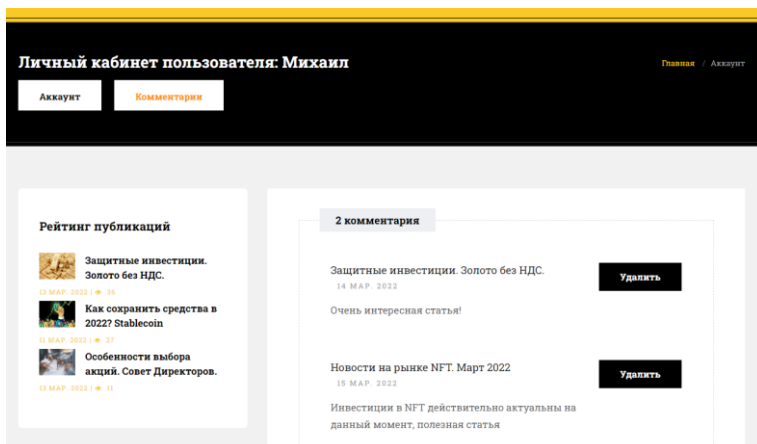


Рисунок 6 – Личный кабинет пользователя. Комментарии

При выборе категории из выпадающего списка навигационной панели открывается страница с отфильтрованными публикациями по выбранной

категории. Кроме того, присутствует функциональность для фильтрации публикаций по тегам. На рисунке ниже представлен пример результата фильтрации публикаций по категории «Руководства».

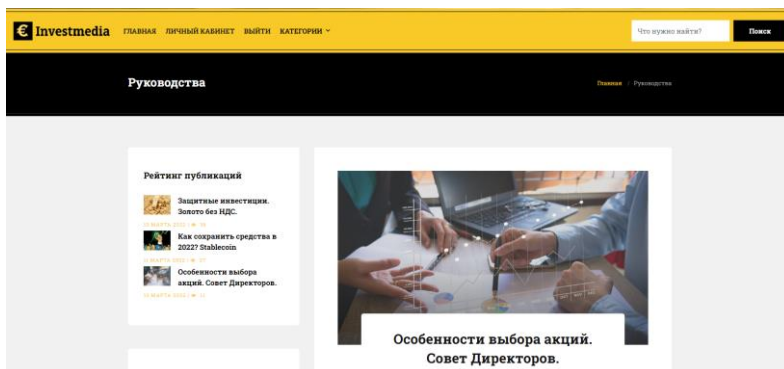


Рисунок 7 – Фильтрация публикаций по категориям

При поиске публикации по ключевым словам (поле для поиска находится в правом верхнем углу навигационной панели) открывается страница с результатами поиска.

После авторизации в системе в роли администратора открывается главная страница панели управления администратора с навигационной панелью в левой части экрана (рисунок 8). На данной странице отображается статистика по публикациям в виде гистограммы и линейной диаграммы с возможностью экспорта графиков в формате «PDF».

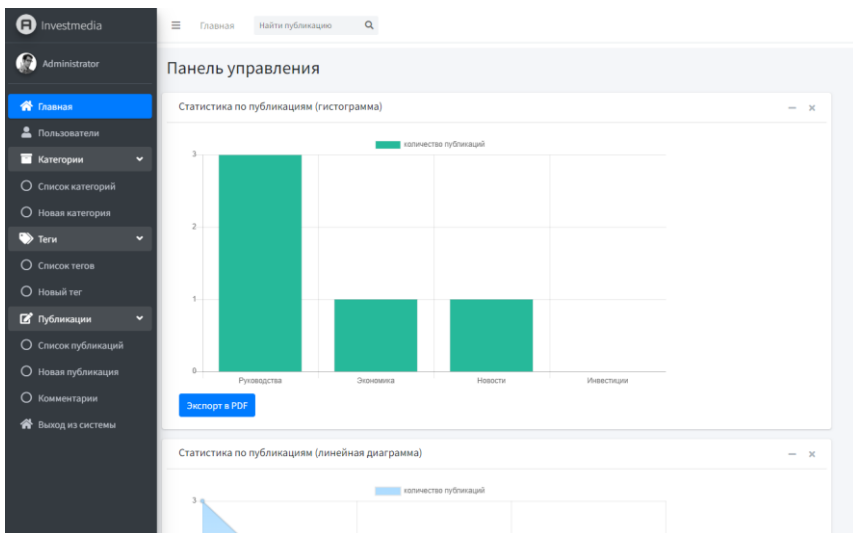


Рисунок 8 – Панель управления. Главная страница

При выборе пункта «Список публикаций» открывается таблица со всеми публикациями и подробной информацией о каждой из них. Последняя колонка таблицы позволяет управлять публикациями (просматривать, редактировать, удалять) – рисунок 9.

Список публикаций

[Добавить публикацию](#) Категория:

#	Наименование	Slug	Категория	Теги	Дата	Комментарии	Управление
25	Защитные инвестиции. Золото без НДС.	zashchitnye-investitsii-zoloto-bez-nds	Экономика	имущество; золото; ндс;	2022-03-13 18:47:27	2	✔ 🔍 ✖
24	Новости на рынке NFT. Март 2022	novosti-na-rynke-nft-mart-2022	Новости	криптовалюта; nft;	2022-03-13 18:17:51	1	✔ 🔍 ✖
21	Как сохранить средства в 2022? Stablecoin	kak-sohranit-sredstva-v-2022-stablecoin	Руководства	стейблкоины; криптовалюта;	2022-03-11 19:41:27	0	✔ 🔍 ✖
22	Как защитить свои деньги и ценные бумаги?	kak-zashchitit-svoi-dengi-i-cennye-bumagi	Руководства	банкротство; брокеры; имущество;	2022-03-13 12:28:38	0	✔ 🔍 ✖
23	Особенности выбора акций. Совет Директоров.	osobennosti-vybora-akcii-sovet-direktorov	Руководства	акции; портфель; развитие;	2022-03-13 13:01:29	0	✔ 🔍 ✖

Рисунок 9 – Панель управления. Список публикаций

Присутствует фильтрация публикация по категориям, записи в таблице отсортированы по убыванию комментариев.

Для создания новой публикации необходимо нажать на кнопку «Добавить публикацию» над таблицей или выбрать соответствующий пункт в навигационной панели. Для создания публикации используется визуальный редактор CKfinder, позволяющий редактировать и оформлять текст по своему вкусу (рисунок 10).

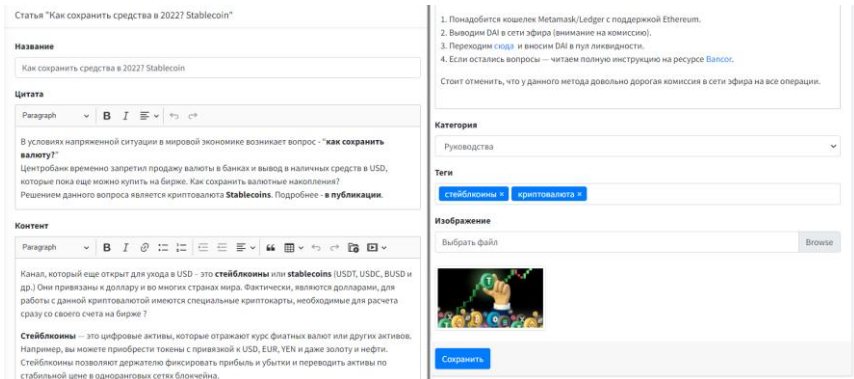


Рисунок 10 – Панель управления. Редактирование публикации

Помимо возможности добавления видео-материалов, таблиц, списков и пр. присутствует функциональность добавления изображений в содержание статьи (используется файловое хранилище для хранения изображений для повторного использования).

Управление категориями и тегами происходит аналогичным образом – для перехода на соответствующие страницы необходимо использовать навигационную панель.

Для модерации комментариев необходимо перейти на страницу «Комментарии», где появится таблица с возможностью удаления комментариев пользователей портала.

Для выхода из системы в роли администратора необходимо нажать на пункт «Выход из системы», после чего неавторизованный пользователь будет переадресован на главную страницу сайта.

Заключение

Таким образом, клиент-серверное приложение «Информационный портал об инвестициях», архитектура и особенности которого рассмотрены в данной статье, полностью реализовано.

Благодаря разработанному portalу пользователь, желающий обучаться в сфере инвестиций и быть в курсе актуальной информации в данной области, экономит время на поиск необходимых данных, используя удобный и интуитивно-понятный интерфейс портала.

Была опрошена определенная выборка людей (возрастная группа от 21 до 49 лет). По шкале usability SUS в виде среднего арифметического между всеми оценками портал удобен и интуитивно-понятен на 92 балла из 100. После опроса были внедрены все рекомендации по улучшению

портала.

Все страницы веб-приложения являются адаптивными и подстраиваются под различные устройства (ПК, ноутбуки, планшеты, смартфоны). Информационный портал не перегружен лишней информацией.

На основе полученного опыта и результатов, в ближайшем будущем планируется расширение функциональности веб-приложения. Например, внедрение статистики в виде графиков, показывающих динамику изменения валютных курсов.

Список литературы

1. Романов, А. А. Конструирование программного обеспечения : учебное пособие / составитель А. А. Романов. – Ульяновск : УлГТУ, 2016. – 126 с.
2. Стаффер, Мэтт. Laravel. Полное руководство [Электронный ресурс]. – Режим доступа: <https://monster-book.com/laravel-polnoe-rukovodstvo> – Загл. с экрана.
3. Суркова Н. Е., Остроух А. В. Проектирование информационных систем [Электронный ресурс]. – Режим доступа: <https://www.litres.ru/av-ostrouh/proektirovanie-informacionnyh-sistem-66011381/> – Загл. с экрана.

DEVELOPMENT OF THE APPLICATION «INVESTMENT INFORMATION PORTAL»

Chernomorchenko A.V. (alinok2000@gmail.com)

Romanov A.A. (romanov73@gmail.com)

Ulyanovsk state technical University, Ulyanovsk

This article represents a description of the developed web application "Investment Information Portal" created to simplify tasks of searching and processing information, providing learning opportunities and quick access to relevant manuals for both beginners and experienced investors.

Keywords: design, information systems, development, software product, investment.

УДК 004.415

**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ
«WEB-ПРИЛОЖЕНИЕ ТОРГОВОЙ ПЛОЩАДКИ
ДЛЯ ЛИМИТИРОВАННЫХ ТОВАРОВ НА ОСНОВЕ
ИНТЕРНЕТ-АУКЦИОНА»**

Шпанцев А.А. (ashpantsev@mail.ru)
Гуськов Г.Ю. (guskovgleb@gmail.com)
Ульяновский Государственный Технический
Университет, Ульяновск

В работе представлено описание разработанной информационной системы «Web-приложение торговой площадки для лимитированных товаров на основе интернет-аукциона». Система предназначена для автоматизации процессов покупки и продажи лимитированных товаров на торговой площадке посредством аукциона.

Ключевые слова: лот, ставка, покупка, аукцион, продажа, лимитированный, заявка, контракт, C#, ASP.NET.

Введение

Во все времена люди обменивались различными товарами. Форм обмена товарами достаточно много, но одна из особенных – аукцион. Аукцион сводит потенциальных покупателей и продавцов вместе, устанавливает конкретную дату совершения сделки, тем самым обостряя конкуренцию между потенциальными участниками.

Самая распространенная форма аукциона - открытый аукцион первой цены с повышением, или английский аукцион. На нем покупатели открыто предлагают свою цену в ответ на предложенную другим участником, при этом каждая следующая ставка выше предыдущей. Предложивший самую высокую цену побеждает и оплачивает ее.

Данный формат проведения продажи товаров или другими словами распределения товаров среди покупателей хорошо подходит для реализации ограниченных партий товара. Товар может быть ограничен в силу естественных причин или искусственных ограничений. К лимитированным товарам относят брендовые вещи, товары от известных

людей, предметы, создаваемые к мероприятиям.

С появлением интернета появилась возможность проведения аукционов на специализированных площадках в дистанционном формате. Они не предполагают прямого взаимодействия между сторонами, что для некоторых людей представляет несомненный плюс.

Требуется разработать интернет-приложение для продажи товаров посредством проведения аукциона: выставления лотов, осуществления торгов, определения победителя и заключения контракта.

Основные бизнес-процессы платформы:

- Управление лотом на торговой площадке
- Процесс модерации лота и процесса продажи
- Проведение аукциона и настройка порядка его проведения
- Заключение эксклюзивного контракта с победителем

Функциональные требования, предъявляемые к системе:

- Регистрация и авторизация в системе (Регистрация возможна в роли пользователя. Регистрацией модератора занимается администратор, который изначально присутствует в системе в единственном числе. Авторизация проходит по адресу электронной почты или имени пользователя, пользователь получает временный токен доступа с правами (пользователь, администратор, модератор));
- Управление лотами (пользователь имеет возможность создавать новый лот, предоставляя полную информацию о лоте, виде аукциона и дате проведения. Все пользователи имеют возможность увидеть выставленные лоты, но только у авторизованных есть право посмотреть полную информацию о лоте. Авторизованные пользователи имеют возможность совершать ставку, покупать лот за полную цену, добавлять лот в список «Избранное», а также удалять из него. В случае, если пользователь просматривает собственный ранее созданный лот, у него есть право его редактировать, а если аукцион имеет закрытый вид, то имеет возможность рассмотрения заявок на участие от других пользователей.)
- Просмотр личного кабинета (авторизованный пользователь, перейдя в личный кабинет, имеет возможность изменять учетные данные, изменять пароль, просматривать поданные заявки, заключать контракты и проводить операции со счётом).
- Проведение модерации (модератор имеет возможность принимать решение о публикации лота: опубликовать или отклонить с указанием причины.)
- Отправка уведомлений в Telegram (авторизованный

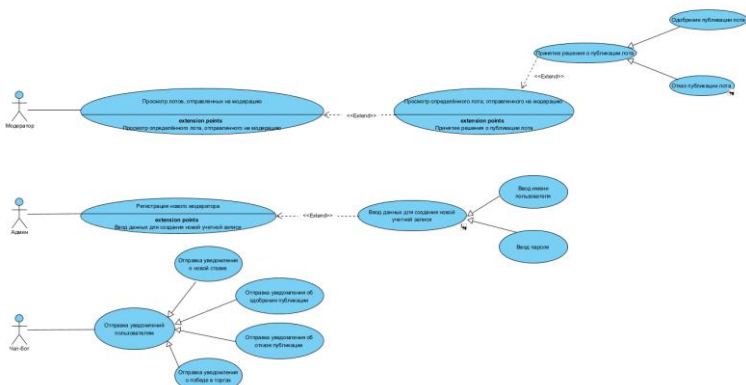


Рисунок 2 – Диаграмма вариантов использования (модератор, администратор, чат-бот)

Разработанные приложения должны быть развёрнуты на локальном веб-сервере или хостинге, доступном для обращения по HTTP-протоколу.

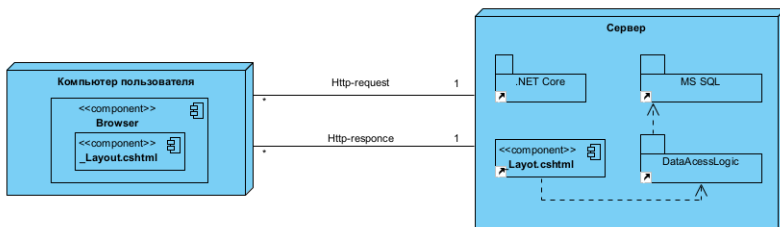


Рисунок 3 – Диаграмма развёртывания системы

Программная реализация системы

Список доступных лотов располагается на главной странице. После прохождения авторизации или регистрации, пользователь автоматически переходит на эту страницу, а также может перейти на эту страницу, нажав кнопку «Главная», которая находится на навигационной панели.




Размещён	Размещён	Размещён
		
Кроссовки Adidas Кроссовки Adidas	Худи Варе Худи Варе	Сумка Gucci Сумка Gucci
Начало торгов: 18.05.2022	Начало торгов: 18.05.2022	Начало торгов: 18.05.2022
Стартовая цена: 3000 руб.	Стартовая цена: 20000 руб.	Стартовая цена: 30000 руб.
Конечная цена: 12000 руб.	Конечная цена: 55000 руб.	Конечная цена: 80000 руб.
Продавец: User2	Продавец: User2	Продавец: User2

Рисунок 4 – Список доступных лотов

Для того, чтоб создать собственный лот, пользователю необходимо перейти в личный кабинет и нажать на кнопку «Создайте свой первый лот». После нажатия на кнопку, пользователю необходимо ввести: название лота, описание лота, тип аукциона, даты начало и окончания подачи заявок (если выбран тип – с ограниченным участием), начальную цену, шаг ставки, конечную цену, даты начала и окончания торгов и выбрать фотографию.

Лот должен пройти модерацию. Доступ к этой странице имеет только модератор. После просмотра, модератор принимает решение об одобрении, либо отказе в публикации. При одобрении лот публикуется, тогда другие пользователи могут совершать ставки или купить лот за конечную цену. При отказе модератор должен указать причину отклонения.

[LA](#) Линейный Аукцион
[Главная](#)
[Правила аукциона](#)

[Личный кабинет](#)
[Избранное](#)
Остаток на счете: 0,00 ₺

Создайте новый лот

Название лота:

На более 500 символов

Описание лота:

На более 500 символов

Тип аукциона:
 Открытый
 С ограниченным участием

Дата начала срока подачи заявок:

Дата окончания срока подачи заявок:

Начальная цена:

Шаг ставки:

Конечная цена:

Дата начала торгов:

Дата окончания торгов:


Фотографии:

Файл размером до 5 Мбайт и сроком отформатирован jpeg, jpg, png

Подтвердить и отправить на проверку

Рисунок 5 – Форма создания лота

[Список лотов](#)
[Правила аукциона](#)



Опубликовать

Отклонить

Укажите причину:


Подтвердить

Рисунок 6 – Форма принятия решения о публикации

После прохождения модерации, лот становится доступным для совершения ставок пользователями. Перейдя на страницу лота, можно совершить ставку, купить за конечную цену и посмотреть историю ставок другими пользователями.

Сервант

Удалить из избранного



До конца: 82 ч. 49 мин. Ставка: 22000 руб. Ставка +2000 руб. Купить 50000 руб.

Информация о лоте

Продавец:	BigDruse
Название:	Сервант
Описание:	Сервант
Дата начала торгов:	19 мая 2022 г.
Дата окончания торгов:	20 мая 2022 г.

История ставок:

Reptized	16.05.2022	22000 руб.
----------	------------	------------

Рисунок 7 – Совершение ставки

Пользователь имеет возможность выбора аукциона с ограниченным участием. При успешном прохождении модерации, другие пользователи не имеют возможности совершать ставки, а только подавать заявки. По истечению времени подачи заявок, продавец принимает решение о допуске к аукциону пользователей, оставивших заявки.

LA Limited Auctions Главная Правила аукциона Личный кабинет Избранное

Остаток на счете: 0,00 Р

Список поданных заявок

Имя пользователя	Допуск до аукциона
User	<input checked="" type="radio"/> Принять заявку
Reptized	<input type="radio"/> Принять заявку
BigDruse	<input checked="" type="radio"/> Принять заявку

Сохранить результат рассмотрения

Рисунок 8 – Список поданных заявок

После окончания аукциона, у пользователя, который сделал последнюю ставку, появляется возможность заключить контракт с продавцом. Подписание контракта включает в себя следующие стадии: подписание участником, подписание продавцом, информация от продавца по доставке, подтверждение получения товара покупателем, после чего

деньги со счёта покупателя списываются и переходят на счёт продавца.

Просмотр контракта

Информация о контракте	
Аукцион:	Сумка Gucci
Цена контракта:	33000,00 руб.
Продавец:	User2
Покупатель:	BigDruse

Статус контракта: Ожидает подписания участником

Подписать

Рисунок 9 – Контракт

Отправкой уведомлений занимается чат-бот. Он имеет возможность отправлять уведомления о: новой ставке, одобрении публикации, победе в торгах, изменении статусов заявок и контракта. В качестве примера отправки уведомлений продемонстрировано уведомление от чат-бота о том, что лот успешно прошёл модерацию.

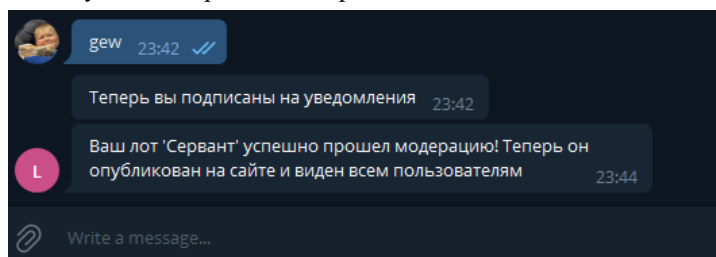


Рисунок 10 – Уведомление об одобрении лот

Заключение

Результатом работы является полностью работоспособное клиент-серверное приложение «web-приложение торговой площадки для лимитированных товаров на основе интернет-аукциона», отвечающее поставленным требованиям.

Разработанное приложение позволит осуществлять проведение аукционов в дистанционном формате, что значительно упростит данный процесс. Помимо этого, дистанционный формат проведения аукциона окажет положительное влияние и на покупателей, так как он не

предполагает прямого взаимодействия между сторонами.

В будущем рекомендуется расширение возможностей приложения. Например, процентное обеспечение ставки, добавление новых типов аукциона: голландский аукцион, аукцион Викри, запрос котировок.

Список литературы

1. Корунова Н.В. Проектирование программного обеспечения: применение на практике / практикум по дисциплине «Проектирование информационных систем» / Н. В. Корунова. – Ульяновск: УлГТУ, 2019. – 69 с. – URL: <http://is.ulstu.ru/disc/pris>. – Режим доступа: для авториз. пользователей.
2. Руководство по ASP.NET Core 5 [Электронный ресурс] – URL: <https://metanit.com/sharp/aspnet5/>
3. .NET Client for Telegram Bot API [Электронный ресурс] – URL: <https://github.com/TelegramBots/Telegram.Bot>

DEVELOPMENT OF INFORMATION SYSTEM «WEB-APPLICATION OF THE MARKETPLACE FOR LIMITED PRODUCTS ON THE BASIS OF THE INTERNET AUCTION»

Shpantsev A.A. (ashpantsev@mail.ru)

Guskov G.Y. (guskovgleb@gmail.com)

Ulyanovsk state technical University, Ulyanovsk

The paper presents a description of the developed information system "Web-application of the trading platform for limited goods based on an online auction". The system is designed to automate the processes of buying and selling limited goods on the trading floor through an auction.

Keywords: lot, bid, purchase, auction, sale, limited, application, contract, C#, ASP.NET.

АВТОРСКИЙ УКАЗАТЕЛЬ

Гладилин Н.Д., 10
Горяйнов А.В., 14
Гуськов Г.Ю., 14, 31, 82
Костин Д.А., 21
Кугузова М.О., 62
Лебедев Д.О., 24
Маринова В.В., 28
Охотников Д.В., 31
Пиков К.А., 37
Романов А.А., 72
Смирнов А.А., 44
Сяман С.В., 52
Тягин И.С., 56
Утицких Д.А., 59
Фаворская М.Н., 62
Хоменко Г.К., 68
Черноморченко А.В., 72
Шпанцев А.А., 82

Научное электронное издание

ПРИКЛАДНЫЕ ИНФОРМАЦИОННЫЕ СИСТЕМЫ (ПИС-2022)

Сборник научных трудов
VIII Всероссийской научно-практической конференции
с международным участием
(г. Ульяновск 30 мая – 05 июня, 2022 г.)

Ответственный за выпуск Е.Н. Эгов

ЛР № 020640 от 22.10.97.

Дата подписания к использованию 11.11.2022.
ЭИ № 1738. Объем данных 3,7 Мб. Заказ № 505.

Ульяновский государственный технический университет
432027, Ульяновск, Сев. Венец, 32.
ИПК «Венец» УлГТУ, 432027, Ульяновск, Сев. Венец, 32.

Тел.: (8422) 778-113
E-mail: venec@ulstu.ru
venec.ulstu.ru