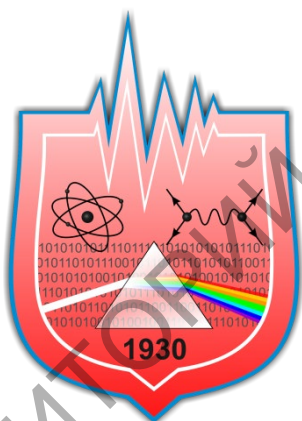


АКТУАЛЬНЫЕ ВОПРОСЫ ФИЗИКИ И ТЕХНИКИ



Гомель
ГГУ имени Ф. Скорины
2022

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный университет
имени Франциска Скорины»

АКТУАЛЬНЫЕ ВОПРОСЫ ФИЗИКИ И ТЕХНИКИ

XI Республиканская научная конференция
студентов, магистрантов и аспирантов,
посвященная 100-летию со дня рождения
академика Белого Владимира Алексеевича

(Гомель, 21 апреля 2022 года)

Сборник материалов

В двух частях

Научное электронное издание

Гомель
ГГУ им. Ф. Скорины
2022

ISBN 978-985-577-769-5

ISBN 978-985-577-770-1 (Ч. 1)

ISBN 978-985-577-771-8 (Ч. 2)

© Учреждение образования
«Гомельский государственный
университет имени
Франциска Скорины», 2022

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный университет
имени Франциска Скорины»

АКТУАЛЬНЫЕ ВОПРОСЫ ФИЗИКИ И ТЕХНИКИ

XI Республиканская научная конференция
студентов, магистрантов и аспирантов,
посвященная 100-летию со дня рождения
академика Белого Владимира Алексеевича

(Гомель, 21 апреля 2022 года)

Сборник материалов

В двух частях

Часть 2

Научное электронное издание

Гомель
ГГУ им. Ф. Скорины
2022

ISBN 978-985-577-769-5

ISBN 978-985-577-771-8 (Ч. 2)

© Учреждение образования
«Гомельский государственный
университет имени
Франциска Скорины», 2022

УДК 530

Актуальные вопросы физики и техники [Электронный ресурс] : XI Республиканская научная конференция студентов, магистрантов и аспирантов, посвященная 100-летию со дня рождения академика Белого Владимира Алексеевича (Гомель, 21 апреля 2022 года) : сборник материалов : в 2 ч. Ч. 2 / М-во образования Республики Беларусь, Гомельский гос. ун-т им. Ф. Скорины ; редкол. : Д. Л. Коваленко (гл. ред.) [и др.]. – Электронные текстовые данные (2,5 МБ). – Гомель : ГГУ им. Ф. Скорины, 2022. – Системные требования: IE от 11 версии и выше или любой актуальный браузер, скорость доступа от 56 кбит. – Режим доступа: <http://conference.gsu.by>. – Заглавие с экрана.

В сборнике рассматриваются актуальные общетеоретические и прикладные проблемы физики и техники.

Издание состоит из двух частей. Во второй части представлены исследования в области новых материалов и технологий; обсуждаются вопросы моделирования физических процессов, использования информационных технологий, в том числе сетевых технологий и СУБД в научных исследованиях, процессах и системах передачи, хранения и защиты информации. Статьи участников конференции размещены в алфавитном порядке.

Издание адресуется научным работникам, преподавателям вузов, аспирантам, магистрантам, студентам, учителям школ, гимназий, колледжей.

Материалы публикуются в соответствии с оригиналом, подготовленным редакционной коллегией, при участии издательства.

Редакционная коллегия:

Д. Л. Коваленко (главный редактор),
А. Л. Самофалов (заместитель главного редактора),
А. А. Серeda (ответственный секретарь),
А. С. Руденков, В. Е. Гайшун, Г. Ю. Тюменков,
Е. А. Дей, А. В. Воружев, П. В. Бычков, Е. Б. Шершнев, С. В. Шалупаев

ГГУ имени Ф. Скорины
246028, Гомель, ул. Советская, 104,
тел. (232) 50-49-03, 50-38-59
<http://www.gsu.by>

© Учреждение образования
«Гомельский государственный университет
имени Франциска Скорины», 2022



Белый Владимир Алексеевич

Ректор учреждения образования «Гомельский государственный университет имени Франциска Скорины» (1969–1973), академик, доктор технических наук, профессор, Заслуженный изобретатель БССР,

Заслуженный деятель науки и техники БССР, лауреат Государственной премии БССР в области науки и техники, автор научного открытия, награжден орденами Ленина, Октябрьской Революции, Трудового Красного Знамени, медалями.

Подробнее [по ссылке](#)

Секция 1 «Новые материалы и технологии»

Председатели:

Руденков Александр Сергеевич, канд. техн. наук, доцент,

Гайшун Владимир Евгеньевич, канд. физ.-мат. наук, доцент.

А. А. Маевский, К. Д. Данильченко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А. В. Семченко**, канд. физ-мат. наук, доцент

ВЛИЯНИЕ ПАРАМЕТРОВ ЗОЛЬ-ГЕЛЬ ПРОЦЕССА НА ФОРМИРОВАНИЕ СЕГНЕТОЭЛЕКТРИЧЕСКИХ ЗОЛЬ-ГЕЛЬ ПЛЁНОК

Золь-гель методом синтезированы порошки и тонкие пленки BST. Было выявлено несколько факторов, влияющих на выбор подходящего метода получения материалов BST, включая стоимость, низкую температуру синтеза, требуемые электрофизические характеристики. Используемое сырье и метод синтеза, технологический процесс влияют на качество полученных материалов BST. Золь-гель метод был использован для синтеза тонких пленок $Ba_{1-x}Sr_xTiO_3$ с различным соотношением Ba и Sr ($x = 0,8; 0,9$).

Приготовлен раствор подходящего объема и концентрации бария и стронция, а 2-метоксиэтанол (2-4 мл) добавляли для стабилизации изопророксида титана (IV) при комнатной температуре. Ацетатную смесь бария/стронций титровали по каплям. Гидроксид аммония, использовался для отрегулирования pH раствора BST до 3-5. Полученные растворы наносили методом центрифугирования на поверхность пластин монокристаллического кремния с платиновым подслоем. После нанесения каждого слоя выполнялся отжиг в муфельной печи в атмосфере воздуха при 800 °C в течение 1 часа [1].

На рисунке 1 показаны зависимости ёмкости сегнетоэлектрической пленки BST от частоты при различных концентрациях x бария/стронция.

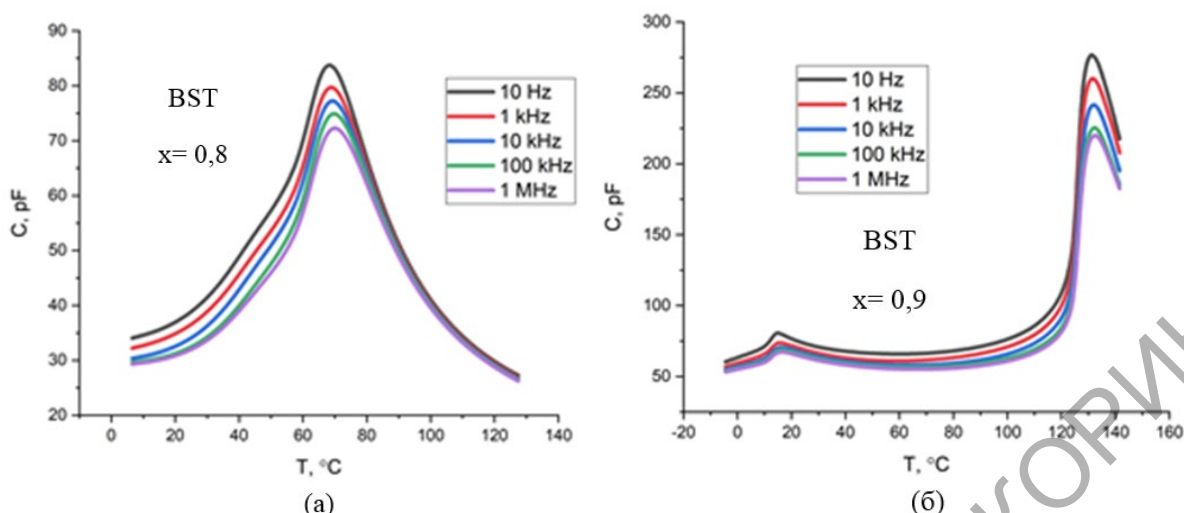


Рисунок 1 – Зависимость ёмкости сегнетоэлектрической пленки BST от частоты при различных концентрациях x бария/стронция: (а) $x = 0,8$ и (б) $x = 0,9$

Наблюдали уменьшение ёмкости с увеличением частоты до 1 МГц, за пределами которой она практически оставалась постоянной. На низких частотах электронные, ионные, дипольная и межфазная / поверхностная поляризация вносят вклад в общее значение диэлектрической проницаемости, но для частот выше 1 МГц вклад межфазной / поверхностной поляризации минимален. При увеличении концентрации бария до $x=0,9$ в тонких плёнках BST температура Кюри сдвигается к $135\text{ }^{\circ}\text{C}$, по сравнению с содержанием бария $x=0,8$ в тонких плёнках BST температура Кюри составляла порядка $70\text{ }^{\circ}\text{C}$. Также было видно, что ёмкость уменьшалась с увеличением температуры выше температуры Кюри. Для увеличения ниже температуры Кюри ёмкость уменьшалась, что приводило к увеличению проводимости. Проводимость увеличивалась с увеличением частоты из-за колебаний решёток, которые приводили к колебаниям атомов или молекул вокруг их узлов, вызванные возбуждением атомов.

Параметр решетки и T_K зависят от отношения Ba/Sr. Температура Кюри зависит от концентрации стронция в BST материале. Сегнетоэлектрическая фаза переходит в параэлектрическую фазу при температуре Кюри. Температура Кюри уменьшалась с ростом концентрации иона Sr^{+} для $\text{Ba}_{0,8}\text{Sr}_{0,2}\text{TiO}_3$ ($70,0\text{ }^{\circ}\text{C}$), $\text{Ba}_{0,9}\text{Sr}_{0,1}\text{TiO}_3$ ($135\text{ }^{\circ}\text{C}$).

Работа выполнена при частичной поддержке ГПНИ «Материаловедение, новые материалы и технологии».

Литература

1. Влияние содержания ниобия на фазовый состав сегнетоэлектрических слоёв $\text{SrBi}_2(\text{Ta}_x\text{Nb}_{1-x})_2\text{O}_9$, синтезированных золь-гель методом / В. В. Сидский [и др.] // Актуальные проблемы физики твёрдого тела : сб. докл. международной научной конференции, ФТТ-2013, Минск, 15–18 октября 2013 г. : в 3 т. / ФТТ; редкол.: Н. М. Олехнович [и др.]. – Минск, 2013. – Т. 2. – С. 267–269.

М. И. Москвичёв

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Д. Л. Коваленко**, канд. физ.-мат. наук, доцент

РАЗРАБОТКА МЕТОДИКИ ПОЛУЧЕНИЯ МИКРОСФЕРИЧЕСКИХ МАТЕРИАЛОВ С ПРИМЕНЕНИЕМ ЗОЛЬ-ГЕЛЬ МЕТОДА И РАСПЫЛИТЕЛЬНОГО ПИРОЛИЗА

Полые микросферы представляют собой неорганические сферические частицы малого размера, диаметр которых варьируется от 0,01 до 1,0 мм. Основными преимуществами микросфер является малая масса, низкая теплопроводность, звукоизоляция, износостойкость, высокая дисперсия, хорошая электроизоляция и теплостойкость, невысокие эксплуатационные затраты, высокую прочность и возможность различного применения [1].

Золь-гель метод является перспективным методом синтеза различных материалов, в том числе, позволяет получать микросферические порошки при низких температурах, по сравнению с традиционными методами получения порошков из твердой фазы.

В методе распылительного пиролиза можно выделить несколько основных этапов: приготовление раствора на основе соединений металлов, распыление и формирование капель аэрозоля, удаление растворителя и дальнейшее протекание химических реакций. Основными преимуществами этого метода являются: возможность получения материалов различного состава и морфологии, экономичность, а также высокая скорость процесса превращения [2].

На рисунке 1 представлены основные этапы разработанной методики формирования микросферических материалов.

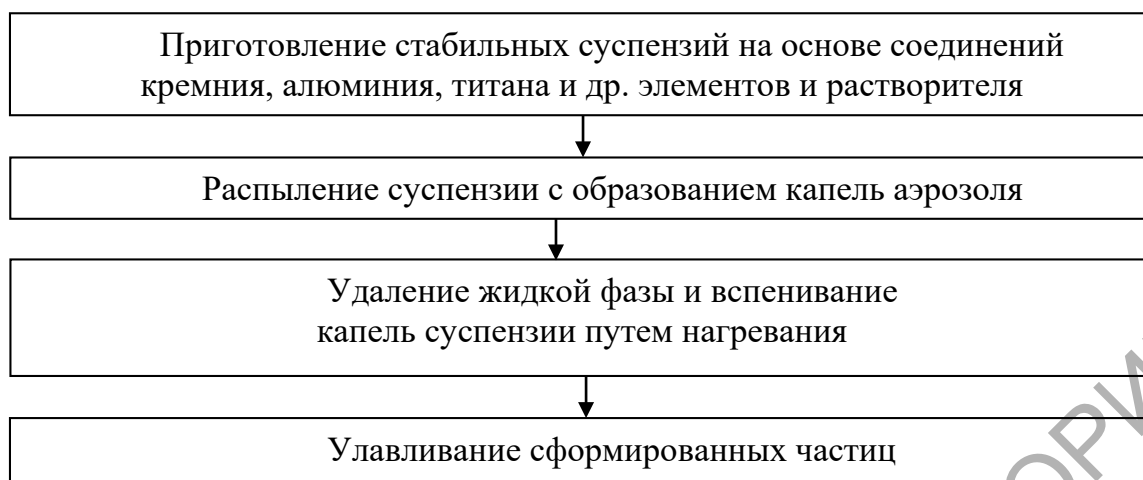


Рисунок 1 – Схема получения микросферических порошков с применением золь-гель метода и распылительного пиролиза

С применением разработанной методики были получены порошки на основе соединений титана и цинка. На начальном этапе были приготовлены следующие водно-спиртовые суспензии:

1. Изопропоксид титана и изопропиловый спирт брали в требуемых пропорциях и тщательно перемешивали.
2. Нитрат цинка, дистиллированную воду и изопропиловый спирт брали в требуемых пропорциях и тщательно перемешивали с использованием ультразвуковой обработки.

Используя метод распылительного пиролиза, приготовленные водно-спиртовые суспензии распыляли на нагревательный элемент установки для получения порошков, где происходило удаление жидкой фазы и вспенивание частиц.

Микроструктура полученных порошков была исследована с помощью металлографического микроскопа серии XJM500 и цифровой камеры Levenhuk M300 BASE. Данный тип микроскопа используется для идентификации и анализа структур различных материалов, в частности, металлов и сплавов на предприятиях, в исследовательских лабораториях и научных центрах. Цифровая камера Levenhuk M300 BASE позволяет передавать изображение с объектива микроскопа на экран компьютера. С помощью камеры и специального программного обеспечения можно детально рассматривать изучаемые образцы и производить обработку изображений.

Микрофотографии порошков на основе титана и цинка представлены на рисунках 2 и 3, соответственно.



Рисунок 2 – Микрофотографии порошка на основе титана

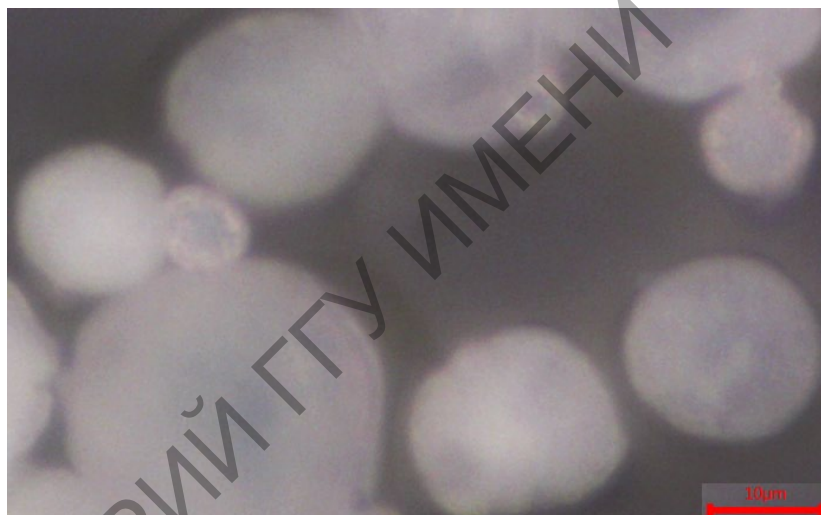


Рисунок 3 – Микрофотографии порошка на основе цинка

Таким образом, с применением золь-гель метода и распылительного пиролиза были получены порошкообразные материалы, имеющие форму, близкую к сферической.

Преимуществами применения золь-гель метода в технологии получения микросферических порошков является сниженная температура синтеза, по сравнению с традиционными методами получения порошков из твердой фазы, возможность использования различных исходных соединений, а также низкие энергозатраты.

Работа выполнена при частичной поддержке ГПНИ «Материаловедение, новые материалы и технологии».

Литература

1. Способ получения полых микросфер: пат. 020857 Рос. Федерация. МПК C04B 38/02 / Цзиньлун Я. [и др.]; заявитель и патентообладатель ХЭБЭЙ ИЛ-БАНДА НЬЮ МАТИРИАЛЗ ЛИМИТЕД КОМПАНИ. – № 201171263; заявл. 20.04.2010; опубл. 27.02.2015.

2. Wang, S. A review of process simulation sand the use of additive sin spray drying / S. Wang, T. Langrish // Food Research International. – 2009. – V. 42. – № 1. – P. 13–25.

П. С. Яночкин¹, К. Д. Данильченко², А. А. Маевский²
(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. ¹**В. Е. Гайшун**, канд. физ.-мат. наук, доцент;

²**А. В. Семченко**, канд. физ.-мат. наук, доцент

ОЦЕНКА И АНАЛИЗ ЭЛЕКТРОФИЗИЧЕСКИХ СВОЙСТВ ЗОЛЬ-ГЕЛЬ ПЛЕНОК НА ОСНОВЕ SiO₂

Введение. В настоящее время интерес к наноматериалам, в том числе и к тонким пленкам, которые используются в различных областях науки и техники, остается стабильно высоким. [1]. Золь-гель метод получения тонкопленочных покрытий привлекателен для исследователей и промышленности тем, что позволяет не только обеспечить требуемые той или иной деятельностью человека свойства и качество покрытий, но и позволяет сократить затраты на процесс формирования покрытий, так как применение золь-гель технологии синтеза материалов является менее энергозатратным по сравнению с другими технологиями и не требует наличия сложного и дорогостоящего оборудования [2].

В ходе данной работы были синтезированы SiO₂ золь-гель покрытия, а также исследованы их электрофизические свойства.

Методика эксперимента и его результаты. Для исследования эффективной диэлектрической проницаемости, токов утечки и анализа вольтамперных характеристик методом центрифугирования были получены кремнеземные покрытия на поверхности монокристаллического кремния с термическим окислом. Все исследования проводились с помощью параметрического анализатора Keithley 4200A-SCS.

На рисунке 1 приведена схема, отражающая структуру слоев исследуемой пластины.

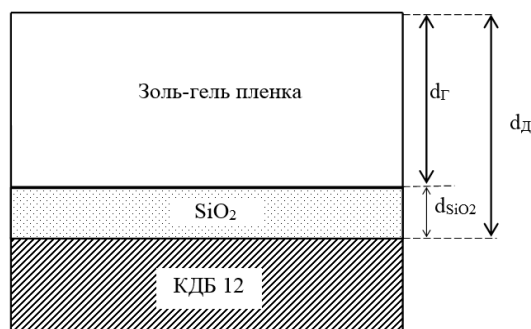


Рисунок 1 – Структура слоев исследуемой пластины

Результаты обработки экспериментальных данных показаны в таблице 1. На рисунке 2 приведены исходные данные измерений с помощью параметрического анализатора Keithley 4200A-SCS.

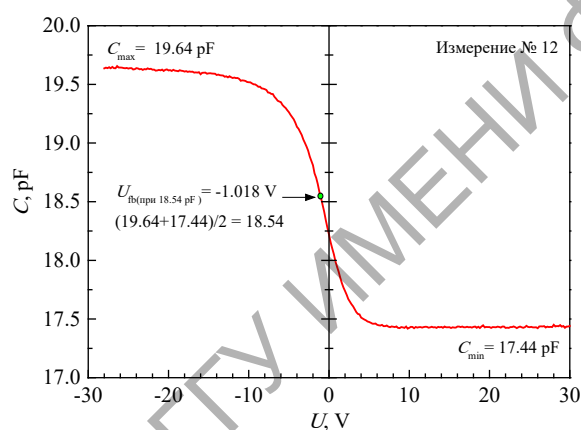


Рисунок 2 – Исходные данные с экрана Keithley для исследуемой пластины

Таблица 1 – Результаты обработки экспериментальных данных

Данные для расчета	
Емкость диэлектрика, C_{\max} (пФ)	19,55
Диэлектрическая проницаемость вакуума, ϵ_0 (Ф/м)	$8,85 \times 10^{-12}$
Площадь контакта металл-полупроводник S (м ²)	$7,29 \times 10^{-7}$
Толщина диэлектрика, d_{Γ} (мкм)	7,8
Толщина слоя SiO_2 , d_{SiO_2} (мкм)	0,05
Диэлектрическая проницаемость SiO_2	3,3
Диэлектрическая проницаемость Si	11,7
Рассчитанные величины	
Эффективная диэлектрическая проницаемость покрытия	2,51
Суммарная емкость диэлектрического слоя, C_d (пФ)	19,64
Емкость плоских зон (из рисунка 2), C_{fb} (пФ)	18,54
Напряжение плоских зон, V_{fb} (В)	-6,0
Напряжение плоских зон (из рисунка 2), V_{fb} (В)	-1,02

В таблице 1 приведено среднее значение рассчитанной диэлектрической проницаемости ϵ (на основании пяти измерений):

– ϵ_d (среднее) – эффективная диэлектрическая проницаемость слоя диэлектрика (слой SiO_2 + золь-гель слой) равна 2,51;

– ϵ_r (среднее) – диэлектрическая проницаемость золь-гель слоя равна 2,47.

Результаты исследований показывают, что величина сдвига напряжения плоских зон V_{fb} составляет $-6,0$ В, что соответствует изменению плотности эффективного заряда на границе раздела, и является небольшой и допустимой величиной.

Результаты экспериментального исследования вольтамперных характеристик представлены на рисунке 3.

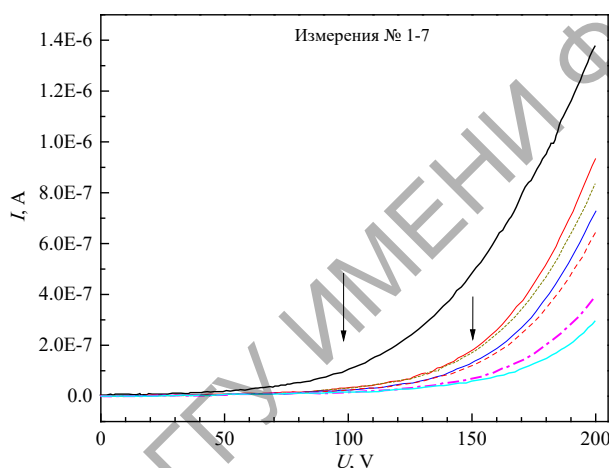


Рисунок 3 – Вольтамперные характеристики

По усредненным результатам серии из семи измерений установлено, что величина тока утечки составляет 1-2 нА до $U=150$ В. Рост тока утечки на рисунке 3 при повышении напряжения, начиная с $U \approx 150$ В (за исключением верхней кривой), соответствует начальному участку пробоя диэлектрического слоя пластины.

Заключение. Сформированы тонкопленочные композиции на основе органических соединений. Выполнено исследование электрофизических свойств части полученных пленок.

Полученные в результате работы данные будут использоваться для дальнейшего изучения электрофизических и структурных свойств тонкопленочных золь-гель покрытий.

Работа выполнена при частичной поддержке ГПНИ «Материаловедение, новые материалы и технологии».

Литература

1. Шабанова, Н. А. Золь-гель технологии. Нанодисперсный кремнезем / Н. А. Шабанова, П. Д. Саркисов. – М. : Бином. Лаборатория знаний, 2012. – 238 с.

2. Кашапов, Н. Ф. Вакуумные технологии нанесения функциональных покрытий / Н. Ф. Кашапов, Г. С. Лучкин, А. Г. Лучкин // Вестник Казан. технол. ун-та. – 2010. – №2. – С. 340–345.

Секция 2 «Моделирование физических процессов»

Председатели:

Тюменков Геннадий Юрьевич, канд. физ.-мат. наук, доцент.

Дей Евгений Александрович, канд. физ.-мат. наук, доцент.

А. В. Павленко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Ю. А. Гришечкин**, канд. физ.-мат. наук, доцент

ПРИБЛИЖЕННОЕ РЕШЕНИЕ ДВУМЕРНОГО УРАВНЕНИЯ ЛОГУНОВА–ТАВХЕЛИДЗЕ В СЛУЧАЕ ПОТЕНЦИАЛА ГАУССА

Уравнение Логунова-Тавхелидзе, описывающее связанные состояния двух скалярных частиц одинаковой массы m , в двумерном импульсном представлении имеет следующий вид:

$$(E^2 - E_p^2) \psi(\mathbf{p}) = \frac{1}{(2\pi)^2} \int V(\mathbf{p}, \mathbf{k}) \psi(\mathbf{k}) \frac{m}{E_k} d^2 \mathbf{k}, \quad E_p = \sqrt{\mathbf{p}^2 + m^2}, \quad (1)$$

где $0 < 2E < 2m$ – энергия двухчастичной системы, \mathbf{p} – относительный импульс в системе центра масс, $\psi(\mathbf{p})$ – волновая функция, $V(\mathbf{p}, \mathbf{k})$ – релятивистский потенциал.

В полярных координатах представим искомую волновую функцию $\psi(\mathbf{p})$ и потенциал $V(\mathbf{p}, \mathbf{k})$ в форме

$$\psi(\mathbf{p}) = \frac{1}{\sqrt{p}} \sum_{\mu=-\infty}^{\infty} \psi_{\mu}(p) \exp(i\mu\varphi), \quad V(\mathbf{p}, \mathbf{k}) = \sum_{\mu=-\infty}^{\infty} V_{\mu}(p, k) \exp(i\mu\gamma), \quad (2)$$

где $\psi_{\mu}(p)$ – парциальная волновая функция, $V_{\mu}(p, k)$ – парциальный потенциал, $p = |\mathbf{p}|$, $0 \leq \varphi \leq 2\pi$ – угол в полярной системе координат,

γ – угол между векторами \mathbf{p} и \mathbf{k} . Подстановка рядов (2) в (1) приводит к интегральному уравнению

$$(E^2 - E_p^2)\psi_\mu(p) = \frac{\sqrt{p}}{2\pi} \int_0^\infty \sqrt{k} V_\mu(p, k) \frac{m}{E_k} \psi_\mu(k) dk. \quad (3)$$

Парциальный потенциал $V_\mu(p, k)$ в импульсном представлении связан с двумерным потенциалом в координатном представлении $V(\rho)$ следующим интегральным соотношением:

$$V_\mu(p, k) = 2\pi \int_0^\infty \rho J_\mu(p\rho) V(\rho) J_\mu(k\rho) d\rho, \quad (4)$$

где $J_\mu(z)$ – функция Бесселя [1], ρ – модуль радиус-вектора в двумерном координатном представлении. В данной работе мы рассматриваем потенциал Гаусса

$$V(\rho) = -\lambda \exp(-a\rho^2), \quad (5)$$

где $\lambda > 0$, $a > 0$ – константы. Подставив (5) в формулу (4) и проинтегрировав [2], получим следующее выражение для парциального потенциала в импульсном представлении

$$V_\mu(p, k) = -\lambda \frac{\pi}{a} \exp\left(-\frac{p^2 + k^2}{4a}\right) I_\mu\left(\frac{pk}{2a}\right), \quad (6)$$

где $I_\mu(x)$ – модифицированная функция Бесселя [1].

Для нахождения приближенного решения уравнения (3) с потенциалом (5) воспользуемся представлением модифицированной функции Бесселя в виде ряда [1]

$$I_\mu\left(\frac{pk}{2a}\right) = \sum_{s=0}^{\infty} \frac{1}{s! \Gamma(\mu + s + 1)} \left(\frac{pk}{4a}\right)^{2s + \mu}. \quad (7)$$

Ограничив ряд (7) некоторым числом первых N слагаемых и подставив полученное выражение в (6), запишем приближенную формулу для потенциала

$$V_{\mu}(p, k) \approx -\lambda \frac{\pi}{a} \exp\left(-\frac{p^2 + k^2}{4a}\right) \sum_{s=0}^N \frac{1}{s! \Gamma(\mu + s + 1)} \left(\frac{pk}{4a}\right)^{2s+\mu}. \quad (8)$$

Потенциал (8) является суммой $N + 1$ сепарабельного потенциала. Подставив (8) в уравнение (3), мы получим следующее равенство:

$$\frac{1}{\lambda} \psi_{\mu}(p) = -\frac{\sqrt{p}}{2a(E^2 - E_p^2)} \exp\left(-\frac{p^2}{4a}\right) \sum_{s=0}^N \frac{1}{s! \Gamma(\mu + s + 1)} \left(\frac{p}{4a}\right)^{2s+\mu} C_{s,\mu}, \quad (9)$$

$$C_{s,\mu} = \int_0^{\infty} \exp\left(-\frac{k^2}{4a}\right) k^{2s+\mu+1/2} \frac{m}{E_k} \psi_{\mu}(k) dk. \quad (10)$$

Умножив (9) на выражение $\exp(-p^2/4a) p^{2j+\mu+1/2} m/E_p$ и проинтегрировав полученное равенство на интервале $p \in [0; \infty)$, мы получим однородную систему линейных алгебраических уравнений для величин $C_{j,\mu}$

$$\frac{1}{\lambda} C_{j,\mu} = -\sum_{s=0}^N M_{s,j} C_{s,\mu}, \quad (11)$$

$$M_{s,j} = \frac{1}{s! \Gamma(\mu + s + 1) 8a^{2s+\mu+1}} \int_0^{\infty} dp \frac{m p^{1+2(j+s)+2\mu}}{E_p (E^2 - E_p^2)} \exp\left(-\frac{p^2}{2a}\right). \quad (12)$$

Вычисление интеграла в (12) выполнено численно по квадратурной формуле прямоугольников [3].

Таким образом, мы заменили приближенно интегральное уравнение (3) алгебраической системой уравнений (11). Найдем теперь решение задачи на собственные значения и собственные векторы для этой системы при разных значениях величины энергии $2E$.

На рисунке 1 приведены графики зависимости энергии двухчастичной системы для первых трех состояний n от величины параметра λ для двух разных значений индекса μ . Сплошной линией обозначены результаты, полученные описанным методом, а штриховой (длинное тире) результаты численного решения уравнения (3) с потенциалом (5).

Численное решение интегрального уравнения найдено методом квадратур прямоугольников [3]. В ходе выполнения вычислений мы полагаем, что $m = 1$, $a = 0,001$, $N = 60$.

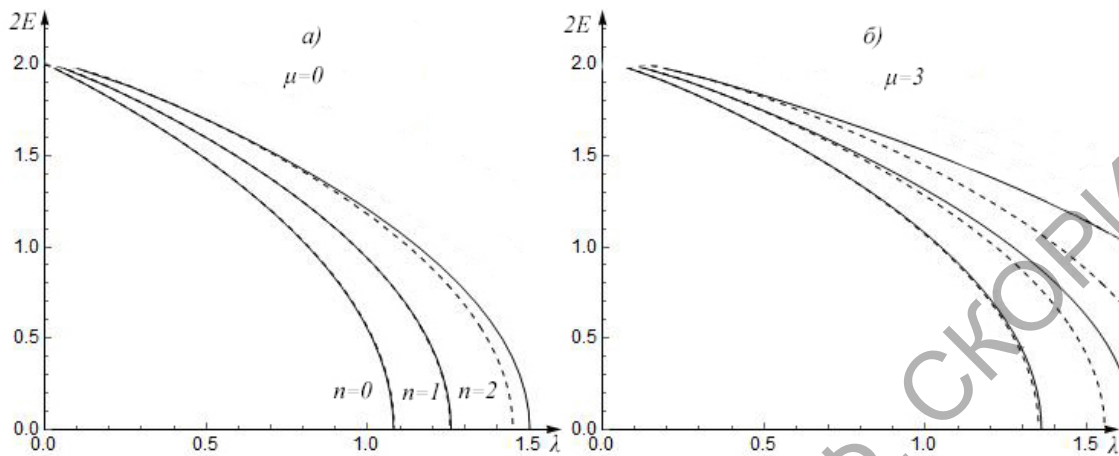


Рисунок 1 – Условия квантования энергии

На рисунке 1 видно, что точность приближения с ростом λ и номера состояния n уменьшается.

На рисунке 2 показаны графики зависимости парциальных волновых функций от переменной p . Нормировка парциальных волновых функции выполнена по формуле

$$\frac{1}{2\pi} \int_0^{\infty} \psi^2(p) dp = 1. \quad (13)$$

На рисунке 2 видно, что число нулей волновой функции равно $n + 1$.

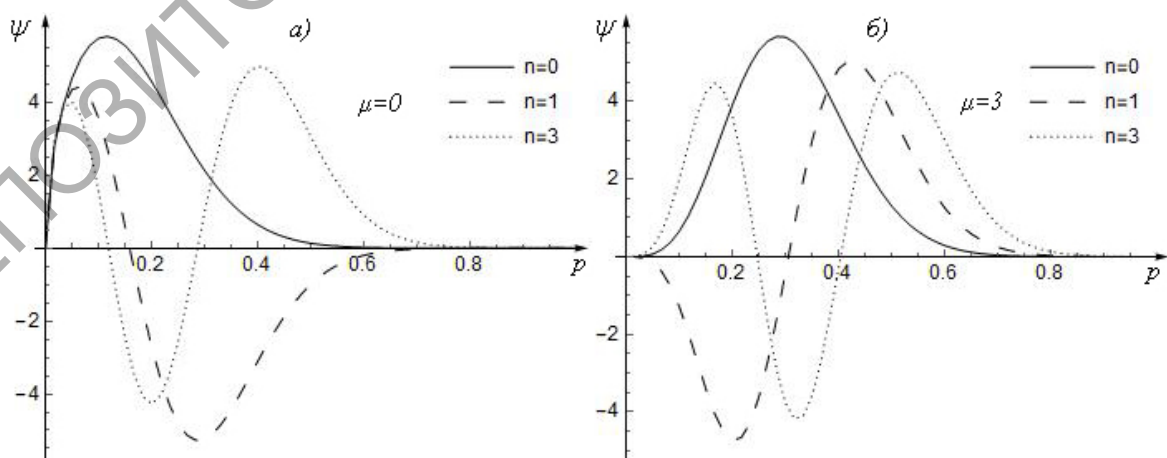


Рисунок 2 – Парциальные волновые функции $\lambda = 1$

В дальнейшем мы планируем выполнить решение рассмотренным в данной работе методом других вариантов релятивистских уравнений с потенциалом (5) и его аналогами.

Литература

1. Arfken, G. Mathematical methods for physicists / G. Arfken, H. Weber, F. Harris. – 7-th ed. – San Diego: Academic Press, 2012. – 1205 p.
2. Градштейн, И. С. Таблицы интегралов, сумм, рядов и производных / И. С. Градштейн, И. М. Рыжик. – Изд. 7-е. – Санкт-Петербург : БХВ-Петербург, 2011. – 1232 с.
3. Калиткин, Н. Н. Численные методы / Н. Н. Калиткин. – М. : Наука, 1978. – 512 с.

Секция 3 «Автоматизация исследований»

Председатели:

Воруев Андрей Валерьевич, канд. техн. наук, доцент.

Бычков Павел Валерьевич, канд. физ.-мат. наук, доцент.

I. A. Ramanau

(Fr. Skorina GSU, Gomel)

Scientific adviser **A. V. Varuyeu**, Ph.D. in technics, associate professor

DEVELOPMENT A MOBILE APPLICATION FOR EDUCATIONAL CONTENT MANAGEMENT

The developed application must ensure stable operation, efficient and fast data processing, as well as the ability to scale.

On this project, we work on this task:

- Development of documentation.
- Compilation of database relationships.
- Creating the design mock-ups.
- Writing the mobile app using React Native.
- Testing application using Expo framework.
- Functional and unit testing the program.
- Writing comments and making code clean.

The project database storage consists of several tables: students, materials, schedule, works, and teachers (Figure 1).

The project was developed using ReactNative platform with ReactJS framework for backend side, AngularJS library for frontend side and H2 as a data storage system. Those tools were chosen for developing the project because of their quality, performance, scalability and other more [1].

To interact with the database, a system is used Prisma framework. Prisma is an open-source ORM that is typically used in backend applications (e.g. API servers) to send queries to a database and resolve data requirements of incoming API requests. Using Prisma to query your database inside of a Server Component saves the roundtrip to the API server by directly loading the data from the database and displaying it in the UI.

Work with the prototype of created application can be seen in Figure 2.

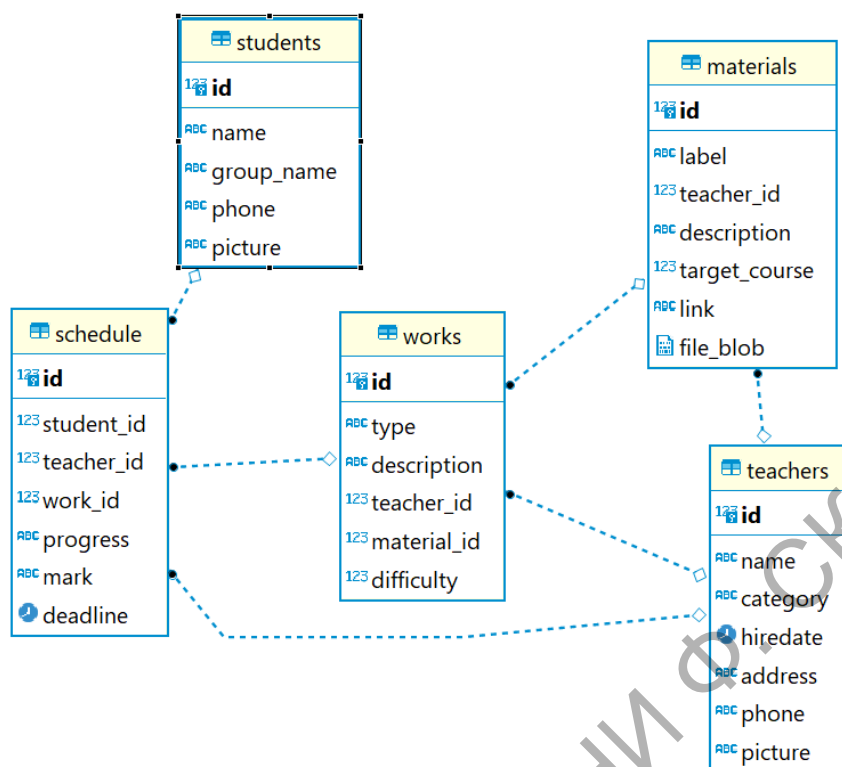


Figure 1 – Database relationships

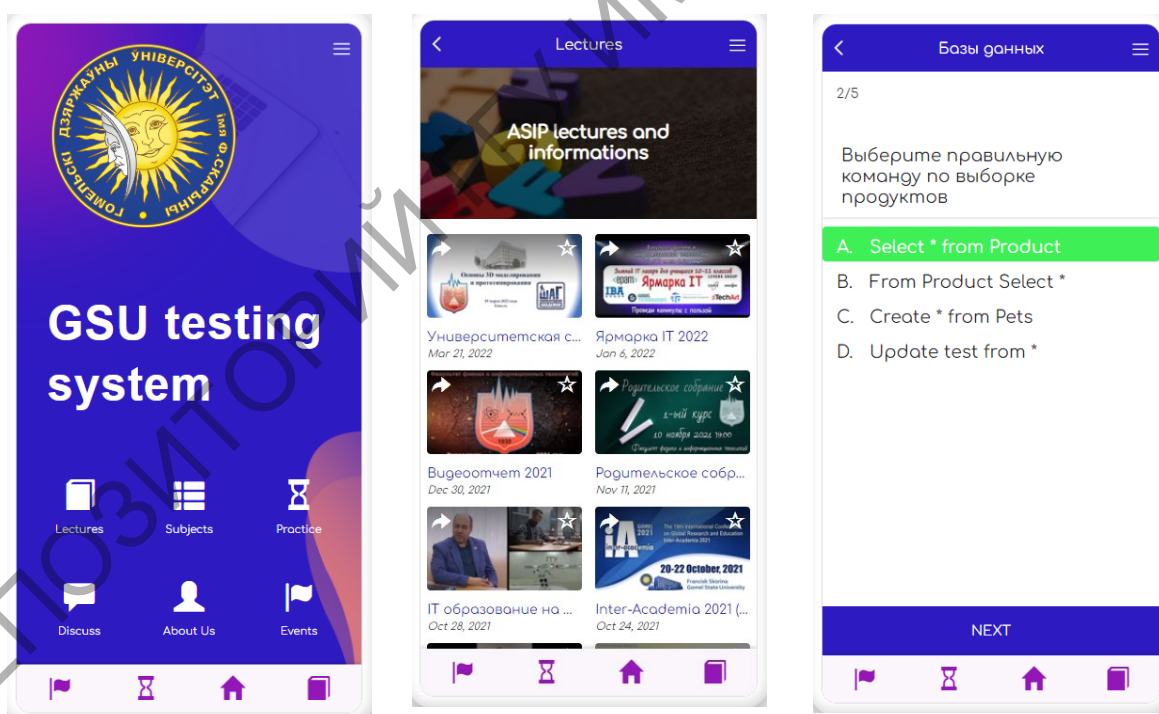


Figure 2 – Database relationships

The correctness of the work is confirmed by successful testing of the developed application.

The result of the study can be assessed as follows:

- 10+ window;
- 40+ React Components Endpoints;
- 14 UI features;
- 96 Unit and Integration tests;
- 5 main and 8 additional tables.

References

1. React Native [Electronic resources]. – URL: <https://reactnative.dev/> – Date of access: 16.03. 2022.

A. A. Zhaldakova

(Fr. Skorina GSU, Gomel)

Scientific adviser **V. N. Levantsov**, senior lecturer

CONTENT MANAGEMENT SYSTEM

The admin panel is usually referred to as a CMS. The content management system (from the English Content Management System) allows you to manage the site, update it regularly, edit and create new pages, articles, products, images. It is also called the «engine» of the site, the speed of content filling; the possibility of quick correction depends on the functionality.

CMS is a separate part of the site, except for the administrator who has access; none of the Internet users can see it. To enter, you need to know the data provided by the developer.

Below you can find examples of most popular CMS.

1 MS Joomla – you can use it free; many prefer it for the possibility of modification in a wide range.

2 CMS WordPress – with its help you can create even a complex project. Open-source code, the ability to integrate plugins, the «transparency» of work incline many to choose it.

3 CMS Bitrix – Russian development, use on a paid basis. Wide functionality, site management using information blocks.

What opportunities does the admin panel give to the owner of the site?

1 Opportunities for resource development, frequent updating of information, addition of relevant data. For online stores, it is indispensable.

2 No need to waste time constantly contacting developers with a request to correct the data.

3 Expanding the site, adding new pages, filling it with fresh, unique content. You cannot do without an admin panel when you need to increase your site in search results, develop a promotion strategy using unique content. The personal participation of the site owner in filling the resource will allow him to achieve his goals faster.

Not always, a site with an admin panel is expensive. A content management system is something that you do not need to save on if you want to work with a resource. Free "engines" or platforms with a small monthly fee may well satisfy the wishes of the customer in terms of their functionality.

A. A. Zhaldakova

(Fr. Skorina GSU, Gomel)

Scientific adviser **V. N. Levantsov**, senior lecturer

THE MAIN GOALS OF BUSINESS CARD SITES, THEIR MAIN ADVANTAGES

The easiest website you can create is a business card website. Its goal is not to sell aggressively, but to gently show all the advantages of the product, leaving the final choice to the visitor. A business card site is a modern business card format that shows your face on the Internet.

First, a business card site serves as a starting project for attracting new customers from the Internet.

1 Having a website for any company is extremely important. Since now, people are looking for any information primarily on the Internet.

2 It is easy to launch advertising campaigns on your site in Google, Yandex, instagram and V Kontakte. In addition, this can affect not only the growth of the company's popularity, but also an increase in income.

3 The ability to quickly change the information on the site so that it remains relevant. This is especially easy to do in the website builder, where it is enough to go to your personal account at any time and replace a couple of lines of text.

4 A business card site can become a kind of online office. Here users will find all the information they need, if necessary, look at the answers to frequently asked questions and leave their data to continue cooperation. Moreover, the installed extensions on the site will immediately send an application to e-mail.

5 The site will help save your advertising budget. With well-tuned SEO settings and the correct filling of the site, it will increase in search results over time. And this means that customers themselves will find it on the Internet, even without additional advertising. In addition, the site's capabilities allow you to place much more information than in an advertising block in a newspaper, booklet, flyer or price list. Thus, you can save on alternative advertising methods.

As you can see, a business card site has many advantages not only for large companies, but also for individual entrepreneurs who are active and want to increase profits. Representation on the Internet will greatly facilitate this task and expand the range of professional activities.

Д. И. Бенько

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ОСНОВНЫЕ МЕХАНИЗМЫ ПОДСИСТЕМЫ УЧЁТА ЗАКУПОК ДЛЯ ТИПОВОЙ КОНФИГУРАЦИИ «1С: ПРЕДПРИЯТИЕ»

Целью создания подсистемы является комплексная автоматизация производственных процессов учёта закупок и контроля складских остатков на предприятии, бухгалтерского, управленческого и оперативного учёта в единой информационной системе в соответствии с нормативно-правовыми актами Республики Беларусь и нормативными документами предприятия.

В подсистеме основная информация хранится в справочниках конфигурации. Так, например, для хранения информации о товарно-материальных ценностях присутствует справочник «Номенклатура», а для хранения информации о поставщиках и покупателях справочник «Контрагенты». Все позиции товарно-материальных ценностей группируются по определенным признакам в группы. Количество вложенных друг в друга групп в типовой версии не ограничено.

Входная информация в системе «1С: Предприятие» представлена документами, предназначенными для описания информации о совершенных хозяйственных операциях или о событиях. Например, документ «Приходная накладная», пользователь создаёт её на основе другого документа – «Интернет продажа», а фактого проведения означает,

что событие, которое данный документ отражает, повлияло на состояние учёта. Документ «Интернет продажа» не является проводимым и служит для отображения информации о товарах, которые необходимо закупить.

Для контроля товарно-материальных ценностей в подсистеме учёта закупок предусмотрен регистр накопления, а также регистр сведений, предназначенный для хранения цен предприятия в разрезе номенклатуры.

Выходная информация в системе «1С: Предприятие» представлена отчётами, предназначенными для обработки накопленной информации и получения сводных данных в удобном для просмотра и анализа виде. В подсистеме складского учёта были созданы отчёты «Товары на складе», «Поставщики», «Покупатели», «Спецодежда на складе».

Д. И. Бенько

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

РАЗРАБОТКА ПОДСИСТЕМЫ УЧЁТА ЗАКУПОК ДЛЯ ТИПОВОЙ КОНФИГУРАЦИИ «1С: ПРЕДПРИЯТИЕ»

Обеспечение эффективного функционирования современного предприятия во многом определяется процессом закупок, оказывающим существенное влияние на устойчивость, эффективность и конкурентоспособность предприятия. Автоматизация закупок приводит к снижению времени обслуживания клиентов и ликвидации избыточных запасов на складах. Возрастает оборачиваемость, а, следовательно, возрастают объёмы реализации и прибыль предприятия.

Целью данной работы является автоматизация процесса закупок товаров для последующей перепродажи их клиентам интернет – магазина. Система должна решать задачи формирования интернет-продаж (заказов) и последующего сбора их в документ для удобной сверки цен, отображённых в счёт – фактуре от поставщика товара.

Среди множества возможных инструментов для разработки подсистемы был выбран программный комплекс «1С: Предприятие 8.3», как наиболее приемлемый и гибкий. Так для рассмотренных в ходе работы программных средств «SAP Ariba», «AVA LRPi», «Precoro» был выявлен ряд недостатков, свидетельствующих о непригодности этих систем для реализации данных задач. В то время как типовая конфигурация

«1С: Предприятие 8.3» – идеально подходящая платформа для реализации проекта, так как создана в точном соответствии с нуждами большинства предприятий и белорусским законодательством.

Подсистема учёта закупок помогает вести закупочную документацию, при сборе и обработке заявок на проведение закупочной деятельности, способствует быстрому формированию отчётов для руководства и для органов власти. Позволяет сохранять и быстро найти необходимую информацию о закупочной деятельности предприятия.

В подсистеме учёта закупок были определены роли для разрабатываемого решения, описаны основные сценарии использования, составлены UML-диаграммы прецедентов, приведена информационно-логическая модель данных и архитектура проекта.

Полученная подсистема имеет достаточный функционал и является отличным средством автоматизации учёта закупок на любом предприятии.

М. В. Бобкова, Д. Д. Троцкая
(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ИСПОЛЬЗОВАНИЕ НЕРЕЛЯЦИОННЫХ БАЗ ДАННЫХ

Альтернативные модели данных стали более распространенными в последние годы. Эти модели нереляционных баз данных стали классифицировать как базы данных NoSQL.

Базы данных NoSQL могут хранить различные типы данных и не должны быть так структурированы, как базы данных SQL. Следовательно, нереляционные базы данных обеспечивают большую адаптивность и гибкость, что делает их более подходящими при работе с большими наборами неструктурированных и несвязанных данных.

Нереляционные базы данных превосходят требования к масштабируемости и доступности и идеально подходят для социальных сетей и приложений реального времени (например, онлайн-игр, обмена мгновенными сообщениями). Технологии NoSQL могут автоматически распределять данные по разным серверам. Это повышает скорость чтения данных в распределенной среде.

В нереляционных базах данных схема данных является динамической и может меняться в любой момент времени. К данным сложнее получить доступ. Такие СУБД отличаются производительностью и

скоростью. Физические объекты в NoSQL обычно можно хранить прямо в том виде, в котором с ними потом работает приложение.

К основным видам NoSQL относятся документно-ориентированные, графовые, колоночные и ключ-значение базы данных.

Согласно рейтингу DB-Engines, в топе самых популярных нереляционных СУБД находятся MongoDB и Redis.

MongoDB – является наиболее широко используемой документно-ориентированной базой данных в мире. Она отличается высокой доступностью, масштабируемостью и безопасностью. Данные хранятся в виде коллекций и документов. Между коллекциями нет сложных соединений типа JOIN. У коллекций не обязательно должна быть схожая структура. У одного документа может быть один набор полей, в то время как у другого документа совершенно другой тип и количество полей.

Redis – представляет собой резидентное хранилище структур данных типа ключ-значение. Поддерживает различные структуры данных: от строк до растровых изображений, потоков и пространственных индексов. В зависимости от настроек Redis может действовать как база данных, кэш или брокер сообщений. Сервер Redis не хранит данные в строго определенных схемах, то есть в таблицах, строках и столбцах. Вместо этого, он хранит данные в структурах данных, что повышает гибкость системы.

Д. В. Богданович

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **М. А. Подалов**, ст. преподаватель

ИЗУЧЕНИЕ СИГНАЛОВ ИНТЕРФЕЙСА IRDA С ПОМОЩЬЮ ЛОГИЧЕСКОГО АНАЛИЗАТОРА

Протокол IrDA (Infra red Data Association) позволяет соединяться с периферийным оборудованием без кабеля при помощи ИК-излучения с длиной волны 880 nm. Порт IrDA позволяет устанавливать связь на коротком расстоянии до 1 метра в режиме точка-точка. IrDA намерено не пытаться создавать локальную сеть на основе ИК-излучения, поскольку сетевые интерфейсы очень сложны и требуют большой мощности, а в цели IrDA входили низкое потребление и экономичность. Интерфейс IrDA использует узкий ИК-диапазон (850–900 nm с 880 nm

«пиком») с малой мощностью потребления, что позволяет создать недорогую аппаратуру и не требует сертификации FCC (Федеральной Комиссии по Связи).

Устройство инфракрасного интерфейса подразделяется на два основных блока: преобразователь (модули приемника-детектора и диода с управляющей электроникой) и кодер-декодер. Блоки обмениваются данными по электрическому интерфейсу, в котором в том же виде транслируются через оптическое соединение, за исключением того, что здесь они пакуются в кадры простого формата – данные передаются 10 bit символами, с 8 bit данных, одним старт-битом в начале и одним стоп-битом в конце данных.

Сам порт IrDA основан на архитектуре коммуникационного COM-порта ПК, который использует универсальный асинхронный приемопередатчик UART (Universal Asynchronous Receiver Transmitter) и работает со скоростью передачи данных 2400–115200 bps. [1]

Логический анализатор представляет собой многоканальное устройство для сбора, хранения и отображения данных, представляющих собой последовательность нулей и единиц (логических состояний), в зависимости от того, больше или меньше входной сигнал заданного порогового уровня. Логические анализаторы делятся на автономные (анализаторы, выполненные в виде единого устройства) и виртуальные (анализаторы, имеющие отдельный блок сбора данных и работающие в связке с ПК по интерфейсу – приставка к ПК).

Логические анализаторы обеспечивают представление собранных данных в виде:

1. Временной диаграммы
2. Последовательности кодов
3. Декодированных кодов (при наличии специального ПО)

Логические анализаторы могут работать синхронном и асинхронных режимах:

1. Синхронный режим (регистрация состояний) - режим сбора данных, при котором момент оцифровки состояний входа логического анализатора, определяется пришедшими из тестируемой схемы тактовыми импульсами на отдельный вход логического анализатора.

2. Асинхронный режим – режим сбора данных, при котором момент оцифровки состояний входа логического анализатора, определяется частотой, задаваемой внутренним генератором логического анализатора. [2]

В данном проекте реализуется анализ входных и выходных сигналов интерфейса IrDA для управления и контроля источника освещения, параметров яркости и цветовой температуры.

Литература

1. IrDA датчик [Электронный ресурс] / Сайт ixbt.com. – Режим доступа: <https://www.ixbt.com/peripheral/irda.html>. – Дата доступа: 21.03.2022.

2. Логический анализатор [Электронный ресурс] / Официальный сайт компании «Контрольно-измерительные приборы и системы». – Режим доступа: https://www.kipis.ru/info/index.php?ELEMENT_ID=20748. – Дата доступа: 21.03.2022.

А. Р. Графов

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А. И. Кучеров**, ст. преподаватель

РАЗРАБОТКА НАБОРОВ ТЕСТОВ ДЛЯ ПРОВЕРКИ ФУНКЦИОНИРОВАНИЯ САЙТОВ

Проект был создан для тестирования правильности работы пользовательского интерфейса веб приложений.

Данный проект дает возможность осуществлять полную проверку работы приложения, статических элементов, веб анимации javascript. Пользователь может тестировать целый ряд элементов страницы: текст, текстовые поля, выпадающие списки, кнопки, чекбоксы (checkboxes), переключатели (radiobuttons). Также пользователь может собирать различные автотесты в тестовые наборы и доставлять их в CI/CD системы, а также наблюдать время выполнения как одного теста, так и тестового набора.

Распечатка подробного отчета, фильтрация, параметризация реализуется с помощью тестового фреймворка pytest.

Оптимизация кода, читабельность и соблюдение принципа DRY достигается использованием фикстур.

В основе архитектуры лежит Page object Model. Эта модель подразумевает использование принципов ООП, в частности наследования. Также в этой модели тесты разделяются по страницам веб приложения, используется принцип «один тест покрывает функционал одной страницы».

В основе любого тестового проекта лежит разработка тестового фреймворка. В тестовом фреймворке создается универсальная основа для создания гибких тестовых наборов. В проекте реализован класс Base, в этом классе описаны все методы, которые могут использоваться

в тестах, в основном методы локализации элементов. Все последующие классы описывают веб-страницы и являются потомками класса Base. Локаторы элементов хранятся в отдельном файле. Фикстуры находятся в файле conftest и отвечают за параметризацию и автоматизацию кода.

Весь проект написан с использованием языка Python, в качестве вебдрайвера используется Selenium, используемая IDE – Pycharm Community Edition.

А. Р. Графов

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А. И. Кучеров**, ст. преподаватель

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ SELENIUM WEBDRIVER ПРИ НАПИСАНИИ АВТОМАТИЗИРОВАННЫХ ТЕСТОВ

Selenium поддерживает автоматизацию всех основных браузеров на рынке с помощью WebDriver. WebDriver – это протокол, определяющий независимый от языка интерфейс для управления поведением веб-браузеров. Каждый браузер поддерживается определенной реализацией WebDriver, называемой драйвером. Драйвер – это компонент, который отвечает за передачу полномочий браузеру и обеспечивающий обмен данными между Selenium и браузером.

Это разделение является частью сознательного усилия, направленного на то, чтобы поставщики браузеров взяли на себя ответственность за реализацию своих браузеров. Selenium по возможности использует эти сторонние драйверы, но также предоставляет свои собственные драйверы, поддерживаемые проектом, для тех случаев, когда это невозможно.

Фреймворк Selenium связывает все эти части вместе с помощью пользовательского интерфейса, который позволяет прозрачно использовать различные серверные части браузера, обеспечивая кросс-браузерную и кроссплатформенную автоматизацию.

Настройка Selenium сильно отличается от настройки других коммерческих инструментов. Прежде чем начать писать код Selenium, необходимо установить библиотеки языковых привязок для выбранного вами языка, браузера, который вы хотите использовать, и драйвера для этого браузера.

При использовании технологии Selenium нужно открыть документацию, посвященную конкретному языку программирования, а также поместить файл вебдрайвера в указанный на сайте каталог.

При использовании технологии Selenium можно локализовать элементы по: id, имени, xpath, гиперссылке, тегу, по имени класса и по CSS селектору.

Также Selenium отвечает за настройку ожиданий. Ожидания необходимы, чтобы скрипт находил элементы вовремя и не завершал работу аварийно.

М. И. Гулевич

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А. И. Кучеров**, ст. преподаватель

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ПОИСКА ОПТИМАЛЬНОГО РЕЖИМА ДЛЯ ФИТНЕС-ТРЕННИНГА

Основная проблема людей, занимающихся спортом – недостаток мотивации и силы воли. Фитнес приложения используются в качестве тренера, а также мотивируют заниматься. Уделяя пару минут в день, пользователь сможет накачать мышцы и подтянуть форму, занимаясь из дома. Человеку не нужны тренера и оборудование – многие упражнения выполняются с собственным весом. Также пользователи встречаются с проблемой, что им трудно разбираться в запутанном интерфейсе приложения.

Мобильные приложения для фитнеса позволяют контролировать активность, путем установки напоминания о тренировках, и сжигание калорий после того, как пользователь завершит упражнение, собирая при этом объемную информацию в профиль пользователя. Чтобы упростить приложение разработчики стараются добавить множество всплывающих подсказок и советов, однако всё равно страницы остаются загруженными.

Для разработки приложения использовалась интегрированная среда разработки Android Studio. При создании приложения «Fitness TyFo» интерфейс создан просто и интуитивно понятно путем вынесения основного меню с тремя пунктами на боковую панель. В главном меню в центре экрана расположились плитки с изображением предмета или действия, с которым связано упражнение, а также его название.

Над плитками добавлена сортировка, чтобы пользователь мог быстро найти определенный род упражнений. Для вовлеченности пользователей заходить в приложение, а также немного отдохнуть между занятиями, создано меню с интересными фактами о спорте, вредной и полезной еде и как правильно соблюдать режим дня.

Цель проекта, чтобы приложение результативно сказалось на здоровом образе жизни людей. Гибкость – одно из самых важных физических качеств человека, о котором многие забывают. Если человек будет развивать данный навык, в будущем ему не придется страдать от коварных заболеваний.

А. А. Дашук

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

РАЗРАБОТКА ПОДСИСТЕМЫ УЧЕТА УСЛУГ ТРАНСПОРТНОЙ ОРГАНИЗАЦИИ

Автоматизация учета услуг транспортной организации необходима для ведения учета различных услуг, предоставляемых транспортной организацией таких как: перевозка грузов, подготовка грузового автомобиля, предоставление грузовой машины на условиях проката или аренды, доставка новых грузовых машин или отремонтированных, транспортно-экспедиционные услуги.

Разработанная подсистема «Автоматизация учета услуг транспортной организации» нужна для ведения полного учета: прихода и расхода денежных средств, путевых листов, кадровых приказов, потребностей в перевозке, приходных и расходных товарных накладных, планирования технического обслуживания, контроль окончания сроков действия документов на автотранспорт, ведение учета ДТП и так далее. Был произведен обзор альтернативных решений данного проекта. Рассмотренные программные обеспечения имеют высокий уровень, но у них есть ряд недостатков, которые препятствуют ведению полного учета. Исходя из результатов обзора было принято решение разработать проект на языке программирования 1С.

В подсистеме «Автоматизация учета услуг транспортной организации» работа осуществляется при помощи следующих объектов конфигурации: константы, функциональные опции, справочники, документы, перечисления, отчёты, регистры сведений и накопления.

Для подсистемы «Автоматизация учета услуг транспортной организации» были определены следующие роли:

- Администратор. Осуществляет администрирование подсистемы, наполняет и создает различные объекты информационной базы;

- Программист. Настраивает подсистему и ее элементы такие как: справочники, документы, регистры, константы, перечисления и т.д. Имеет возможность изменять, добавлять и просматривать все объекты конфигурации.

Данная подсистема легка в освоении, проста в использовании и интуитивно понятная для простого пользователя, она осуществляет: контроль приходных и расходных товарных накладных; создание путевых листов; оформление потребностей в перевозке; учет прихода и расхода денежных средств; просмотр отчетной информации; учет транспортных средств и сведений о них; ведение договоров с клиентами и поставщиками; расчет персональных скидок клиентов; планирование технического обслуживания и т. д.

Данная подсистема дает возможность обрабатывать большие объемы информации, предоставляет удобный интерфейс для работы с данными, позволяет пользователю, как просматривать необходимую информацию, так и, по мере необходимости, манипулировать ею, например, добавлять, изменять, удалять информацию и т.д., что значительно упрощает процесс ведения учета в информационной базе.

А. А. Дашук

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

АВТОМАТИЗАЦИЯ УЧЕТА УСЛУГ ТРАНСПОРТНОЙ ОРГАНИЗАЦИИ

Разработка подсистемы «Автоматизация учета услуг транспортной организации» происходила на базе нетиповой конфигурации 1С, так как была написана полностью с нуля.

Входная информация подсистемы представлена документами и справочниками:

- документ «Потребность в перевозке» содержит информацию об организации, контрагенте и его договоре, FTL-перевозка или LTL. Табличная часть «Пункты назначения» в которой хранится информация о

виде доставки, адресе отправления и назначения, маршруте, виде перевозки и промежуточных пунктов. В табличной части «Товары к перевозке» находится информация о перевозимом товаре, его количестве, цене и сумме. А также полная информация о параметрах отправления и прибытия таких как: отправитель, контактное лицо, время отправления и прибытия, количество часов, проведенных на стоянке;

– документ «Расходная накладная» содержит информацию о валюте, контрагенте, договоре, сотруднике, который оформляет документ. В табличной части «Товары» хранится информация о номенклатуре и ее количестве, цене и сумме, скидке на товар и сумме скидок, общая сумма. Табличная часть «Услуги» содержит данные о предоставляемых услугах и их количестве. Во вкладке «Прочее» находится поле «Комментарий» для пометок и записей;

– справочник «Транспортные средства» содержит в себе название модели транспортного средства, вид перевозки, госномер, дату выпуска, объём двигателя, расход топлива, вид топлива, грузоподъемность, вместимость, ширина, высота и глубина. Имеется возможность перейти по гиперссылке из формы в справочник «Марки транспортных средств» и создать новую марку. Так же есть функционал по созданию на основании данного транспортного средства его комплектации;

– справочник «Контроль окончания сроков действия документов» включает в себя данные о водителе, его водительском удостоверении, о транспортном средстве на котором водитель работает и проходит техосмотр, страхование транспортного средства и медицинский осмотр водителя и так далее.

После того, как вся информация будет внесена в информационную базу, можно создавать отчеты, которые являются выходными данными подсистемы «Автоматизация учета услуг транспортной организации».

Отчет «Оплаты» содержит подробную информацию о поступлении денежных средств и валюте, клиенте от которого поступили средства, дате поступления, договоре по которому произошло поступление, способе оплаты (наличные или безналичные), сумме поступления.

Отчет «Расходные накладные» содержит информацию о документе, который произвел расход товара. В нём присутствуют такие данные как номенклатура, которая списывается, дата списания, контрагент, который приобрел товар, договор по которому прошла покупка, количество номенклатуры и общая сумма документа.

В. В. Дейниченко
(ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **А. И. Кучеров**, ст. преподаватель

ТЕСТИРОВАНИЕ ВЕБ-ПРИЛОЖЕНИЯ

Тестирование веб-приложения – это процесс оценки и проверки того, что веб-приложение выполняет все свои функции. Преимущества тестирования включают выявление ошибок, снижение финансовых затрат на разработку веб-приложения и повышение его эффективности и производительности.

Существует множество различных типов тестов веб-приложения, каждый из которых имеет конкретные цели и стратегии:

Приемочное тестирование: Проверка того, работает ли вся система должным образом.

Интеграционное тестирование: обеспечение совместной работы программных компонентов или функций.

Модульное тестирование: Проверка того, что каждый программный модуль работает должным образом. Модуль – это самый маленький тестируемый компонент приложения.

Функциональное тестирование: Проверка функций путем эмуляции бизнес-сценариев на основе функциональных требований. Тестирование черного ящика – это распространенный способ проверки функций.

Тестирование производительности: Тестирование того, как программное обеспечение работает при различных рабочих нагрузках. Нагрузочное тестирование, например, используется для оценки производительности в реальных условиях нагрузки.

Регрессионное тестирование: Проверка того, нарушают ли новые функции или ухудшают функциональность. Тестирование работоспособности может использоваться для проверки меню, функций и команд на поверхностном уровне, когда нет времени для полного регрессионного теста.

Стресс-тестирование: Проверка того, какую нагрузку может выдержать система, прежде чем она выйдет из строя. Считается разновидностью нефункционального тестирования.

Тестирование удобства использования: Проверка того, насколько хорошо клиент может использовать систему или веб-приложение для выполнения задачи.

В каждом случае проверка базовых требований является критической оценкой. Не менее важно и то, что исследовательское тестирование помогает тестирующему или команде тестирования выявлять трудно прогнозируемые сценарии и ситуации, которые могут привести к ошибкам в программном обеспечении.

Для разработки тестирования сайта abiturient.gsu.by был выбран такой язык как JavaScript и TypeScript. Программная платформа Node.JS и инструмент тестирования Selenium WebDriver, а также написание автоматизированного фреймворка тестирования веб-приложения.

В. В. Дейниченко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **А. И. Кучеров**, ст. преподаватель

РАЗРАБОТКА ФРЕЙМВОРКА ДЛЯ ТЕСТИРОВАНИЯ ВЕБ-ПРИЛОЖЕНИЯ

Фреймворк – это реальные или концептуальные структуры, предназначенные для того, чтобы служить поддержкой или руководством для создания чего-то, что расширяет структуру до чего-то полезного.

В компьютерных системах фреймворк часто представляет собой многоуровневую структуру, указывающую какие программы, могут или должны быть созданы и как они будут взаимодействовать.

Фреймворк тестирования – это не отдельный инструмент или процесс, а набор инструментов, работающих вместе для поддержки автоматического тестирования любого приложения. Он объединяет различные функции, такие как библиотеки, тестовые данные и различные повторно используемые модули. Фреймворк поддерживает автоматизированное тестирование в качестве руководства по технической реализации.

Последовательные циклы разработки веб-приложения требуют многократную работу по набору ручных тестов. Используя фреймворк для тестирования, можно записать этот набор тестов и воспроизвести его в режиме реального времени. Как только, набор ручных тестов автоматизирован, вмешательство специалиста по тестированию не требуется.

Цель фреймворка для тестирования состоит в том, чтобы сократить большой объем тестов, выполняемых вручную, а не полностью исключить его.

В процессе автоматизации тестирования веб-приложения, выполняются следующие шаги:

Выбирается инструмент тестирования, который зависит от технологии, на которой построено веб-приложение.

Область автоматизации – это область тестируемого веб-приложения, которая будет автоматизирована специалистом.

Планирование, проектирование и разработка. На этом этапе создается план и стратегия написания фреймворка для тестирования.

Выполнение может быть выполнено с помощью средства автоматизации напрямую, или через средство управления тестированием, которое вызовет средство автоматизации.

Подход к обслуживанию автоматизации тестирования. На этом этапе тестирования автоматизации, проводится проверка, чтобы удостовериться в работе новых функций, которые были добавлены в веб-приложение.

Для разработки фреймворка автоматизированного тестирования сайта `abiturient.gsu.by` был выбран такой язык как JavaScript и TypeScript. Программная платформа Node.JS и инструмент тестирования Selenium WebDriver.

А. С. Дробышевский

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Кулинченко**, ст. преподаватель

СОЗДАНИЕ И НАСТРОЙКА LINUX АГЕНТА ДЛЯ CI/CD СИСТЕМЫ TEAMCITY

Агент сборки TeamCity – это часть программного обеспечения, которая прослушивает команды с сервера TeamCity и запускает фактические процессы сборки.

Для дипломного проекта было необходимо автоматизировать сборку и доставку Docker контейнеров. Данную функцию можно было реализовать на существующем сервере TeamCity, установленной на основе операционной системы Windows. Но нюансом послужил тот факт, что проектировался этот сервер точно под нужды сборки и доставки основного приложения, вследствие чего при добавлении дополнительного программного обеспечения ощущалась нехватка оперативной памяти. Была возможность увеличить размер основной виртуальной машины с 8 Гб оперативной память до 16 Гб, но это в свою очередь сильно увеличивало стоимость обслуживания сервера. Последующие расчеты показали, что при использовании отдельного выделенного

сервера на основе Linux появлялась возможность экономии в 8-ми кратном размере.

Настройка созданного сервера под управлением ОС Linux приведена ниже.

Первым шагом будет подготовка операционной системы к обновлениям и установке Java.

```
sudo apt update
```

```
sudo apt install default-jre -y
```

Далее необходимо указать глобальную переменную JAVA_HOME.

```
JAVA_HOME="/usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/java"
```

Следующие скрипты предназначены для скачивания изначальных ресурсов и кода агента, с которым и будут работать в последствии.

```
sudo apt install unzip
```

```
mkdir teamcity-agent
```

```
wget https://teamcity.projectdomain.com/update/buildAgentFull.zip --no-check-certificate
```

```
mv buildAgentFull.zip teamcity-agent/
```

```
cd teamcity-agent/
```

```
unzip buildAgentFull.zip
```

```
rm -rf buildAgentFull.zip
```

После того, как были скачаны ресурсы на виртуальную машину, начинается шаг настройки агента. Для этого необходимо перейти в директорию conf и отредактировать параметр serverUrl со значения http://teamcity.com:8111 на необходимое – домен основного TeamCity сервера, с учетом https соединения.

```
cd conf/
```

```
cp buildAgent.dist.properties buildAgent.properties
```

```
nano buildAgent.properties
```

```
#change serverurl to https://teamcity.projectdomain.com
```

Для обеспечения защищенного соединения необходимо установить сертификаты и связать основной TeamCity сервер и агент по https.

Создать в директории conf директорию ssl, в которую поместить сертификат ser. Помещение сертификата производится с помощью подключения к виртуальной машине через WinSCP и помещением сертификата в директорию ssl.

```
mkdir ssl
```

Далее необходимо добавить данный сертификат в доверенные с помощью утилиты keytool, поставляемой вместе с java. Запускать следует с помощью учетной записи root.

```
sudo su
```

```
/usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/keytool -importcert -file  
/home/azureuser/teamcity-agent/conf/ssl/procomplynet.cer -srcstoretype  
pkcs12 -keystore /usr/lib/jvm/java-1.11.0-openjdk-  
amd64/lib/security/cacerts
```

После данной настройки агент необходимо запустить следующей командой.

```
~/teamcity-agent/bin/agent.sh start
```

В последствии агент подключится к основному серверу и останется его авторизовать в веб интерфейсе TeamCity.

Далее надо установить необходимое программное обеспечение для сборки Docker контейнеров.

```
sudo apt-get update  
sudo apt-get install \  
ca-certificates \  
curl \  
gnupg \  
lsb-release
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --  
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

```
echo \  
"deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/key-  
rings/docker-archive-keyring.gpg] https://download.docker.com/linux/ub-  
untu \  
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list >  
/dev/null
```

```
sudo apt-get update  
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

Протестировать работоспособность Docker можно с помощью следующей команды.

```
sudo docker run hello-world
```

Следующей командой авторизоваться в Docker Registry

```
docker login projectregistry.azurecr.io  
#use credentials from azure registry
```

Последним шагом для реализации сборочной конфигурации является установка Azure cli, с помощью которой контейнеры будут доставляться на вычислительные мощности.

```
curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash
```

Остается только залогиниться.

az login

Итогом данной работы является отдельно выделенный Linux агент, работающий по безопасному соединению, на который установлено необходимое программное обеспечение для сборки и доставки Docker контейнеров на вычислительные мощности.

А. А. Зубрицкий

(Институт физики НАН Беларуси, Минск)

Науч. рук. **М. С. Усачёнок**, канд. физ.-мат. наук

ПРИМЕНЕНИЕ СРЕДСТВ ЯЗЫКА PYTHON ДЛЯ ОБРАБОТКИ ДАННЫХ, ПОЛУЧЕННЫХ СВЧ РЕЗОНАТОРНЫМ МЕТОДОМ

Введение

СВЧ резонатор является одним из основных средств диагностики газоразрядной плазмы, служащий для определения плотности и частоты столкновений электронов. Плотность электронов определяется по смещению резонансной кривой резонатора, а частота столкновений по изменению его добротности. В нашей предыдущей работе [1] мы рассмотрели возможность создания автоматизированной системы регистрации плотности электронов на основе платы National Instruments PCIe-6351. Данная работа посвящена обработке получаемых данных с помощью средств среды программирования Python, как одного из наиболее бурно развивающихся языков программирования.

Экспериментальная установка

Экспериментальная установка представляла собой 10-см СВЧ резонатор, портативный векторный анализатор спектра N9918A (Keysight) и ПК. Цилиндрический резонатор (диаметр – 9 см, высота – 3 см) и без неоднородностей имеет центральную резонансную частоту 2525,4 МГц и добротность 765. Исследуемая газоразрядная плазма создавалась в лампе ГШ-5 (газ неон при пониженном давлении, рисунок 1, а) при пропускании через нее постоянного тока порядка единиц миллиампер. Зарегистрированные резонансные контуры для нескольких значений разрядного тока представлены на рисунке 1, б.

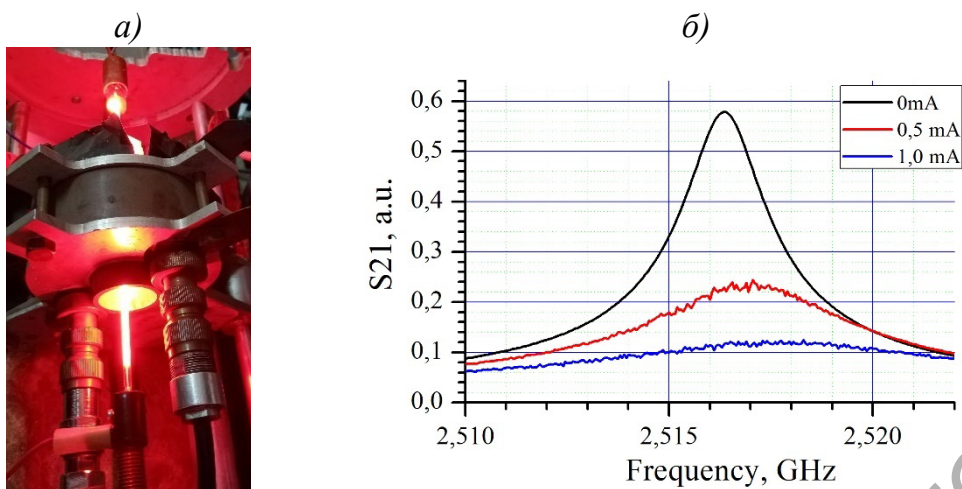


Рисунок 1 – а) Фото 10-см СВЧ резонатора с лампой ГШ-5; б) резонансные контуры для нескольких значений разрядного тока лампы ГШ-5

Средства обработки данных в Python

Как видно, резонансный контур очень близок к Гауссову контуру и может быть им аппроксимирован, то есть с помощью средств Python решалась тестовая задача подбора центральной частоты резонаторного контура и вычисления его добротности с целью дальнейшей оценки частоты столкновений и плотности электронов в плазме ламп ГШ-5 [2]. Пример результата подбора контуров представлен на рисунке 2.

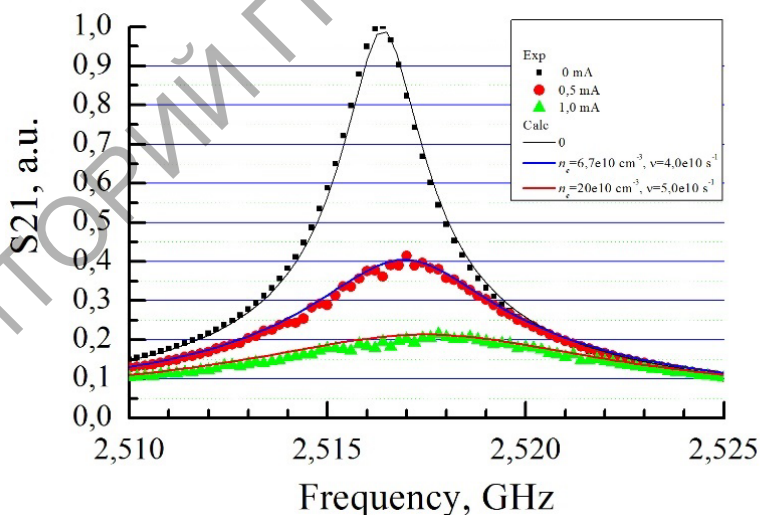


Рисунок 2 – Пример сопоставления экспериментальных и расчетных резонансных контуров

Для достижения поставленной цели была написана программа позволяющая открывать csv-файлы, получаемые от анализатора спектра и

вычислять ne и v , и были использованы следующие основные средства Python [3, 4]:

Matplotlib – один из самых популярных пакетов Python, используемых для визуализации данных в 2D графики, который применяет NumPy, числовое математическое расширение Python. Он предоставляет объектно-ориентированный API, который помогает встраивать графики в приложения;

NumPy, что означает «Числовой Python», – это библиотека, состоящая из объектов многомерного массива и набора процедур для обработки этих массивов. Используя NumPy, можно выполнять математические и логические операции над массивами. Matplotlib вместе с NumPy можно рассматривать как эквивалент MATLAB с открытым исходным кодом;

SciPy – научная библиотека для Python, является библиотекой с открытым исходным кодом, лицензированной BSD для математики, науки и техники. Библиотека SciPy зависит от NumPy, который обеспечивает удобную и быструю манипуляцию с N-мерным массивом.

Подобную программу можно написать и, например, в Mathematica, но есть существенное отличие, состоящее в том, что в Python кроме математического аппарата возможна реализация индивидуального интерфейса для пользователя.

Выводы

В результате выполнения данной работы на языке программирования Python и с помощью его средств реализована программа обработки резонансных кривых 10-см СВЧ резонатора, позволяющая производить оценку плотности и частоты столкновений электронов в плазме газового разряда.

Работа выполнена при поддержке гранта БРФФИ-РФФИ T21PM-120.

Литература

1. Зубрицкий А. А. Применение платы National Instruments для автоматического определения плотности электронов СВЧ резонаторным методом [Электронный ресурс] / А. А. Зубрицкий, М. С. Усачёнок / Актуальные вопросы физики и техники : X Респуб. научн. конф. студентов, магистрантов и аспирантов : материалы: в 2 ч. Ч. 1 (Гомель, 22 апр. 2021 г.) / ГГУ им. Ф. Скорины ; редкол. : Д. Л. Коваленко (гл. ред.) [и др.]. – Гомель, 2021. – С. 107–110.

2. Голант, В. Е. Сверхвысокочастотные методы исследования плазмы / В. Е. Голант. – М. : Наука, 1968. – 328 с.
3. Дауни, А. Б. Цифровая обработка сигналов на языке Python. / А. Б. Дауни. – М. : ДМК Пресс, 2017. – 160 с.
4. Любанович, Б. Простой Python. Современный стиль программирования. / Б. Любанович. – 2-е издание. – СПб. : Питер, 2022. – 592 с.

И. А. Киёк

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ НОВОСТЕЙ CYBERNEWS

Новости существенным образом влияют на нашу повседневную жизнь, и поэтому необходимо проявлять к ним особый интерес, чтобы постоянно быть в курсе тех изменений, которые происходят в жизни.

Новостные ценности, также называемые новостными критериями, определяют охват освещения новости средствами массовой информации, а также интерес, проявляемый к новости аудиторией. Новостные ценности не универсальны и могут значительно отличаться в зависимости от культур.

Быть в курсе событий – ключ к успеху в нашем быстро меняющемся мире. Очень важно найти приложение, которое поможет своевременно получать нужные новости. Интернет изменил способ просмотра новостей. Вместо того, чтобы позволить телевизионному продюсеру или редактору газеты определять самое важное для нас, а затем покупать их продукт, мы теперь можем свободно бродить по волнам Интернета в поисках новостей, которые наиболее важны для нас. Есть масса приложений, которые доставляют новости, и отслеживать их все может быть сложно.

В основе программной части лежит интегрированная среда разработки производства Google, с помощью которой разработчикам становятся доступны инструменты для создания приложений на платформе Android OS – Android Studio [1]. Для хранения настроек, информации о статистике проведения турниров используется база данных SQLite. SQLite – это система управления базами данных, отличительной особенностью которой является ее встраиваемость в приложения. Это значит, что большинство СУБД являются самостоятельными приложениями, взаимодействие с которыми организовано по принципу клиент-сервер.

Целью проекта является предоставление актуальной и качественной информации для пользователя, а также возможность отслеживать игровые события в режиме реального времени. В настоящее время важно иметь последнюю информацию и грамотно ею распоряжаться.

Литература

1 Web Documentation [Электронный ресурс]. – URL: <https://developer.android.com/docs>. – Дата доступа: 17.03.2022.

И. А. Киёк

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

АРХИТЕКТУРА ВЕБ-ПРИЛОЖЕНИЯ НОВОСТЕЙ CYBERNEWS

Одними из элементов моделей данных в проекте является взаимодействие клиента и базы данных с приложением, которые представлены в виде диаграммы вариантов использования (рисунок 1). Суть данной диаграммы состоит в следующем: проектируемая система представляется в виде множества актеров, взаимодействующих с системой с помощью так называемых вариантов использования.

Для рассмотрения взаимодействия между компонентами мобильного приложения для просмотра новостей киберспорта была спроектирована диаграмма последовательности для варианта «Посмотреть статью». Схема последовательности для варианта использования «Посмотреть статью» представлена на рисунке 2. В этой схеме отображены четыре объекта разрабатываемого мобильного приложения: «Интерфейс», класс главной страницы «MainPage», класс настроек «Settings», класс уведомлений «Toast» и одно действующее лицо – «Пользователь».

Работа мобильного новостного приложения начинается с выбора пользователем опции «Посмотреть статью» на главной странице. Согласно диаграмме последовательности, приложение ожидает действия пользователя. С помощью интерфейса пользователь открывает статью. В это время мобильное приложение загружает настройки, после отображает новость. Пользователь после прочтения статьи может нажать на специальный символ для закрытия окна.

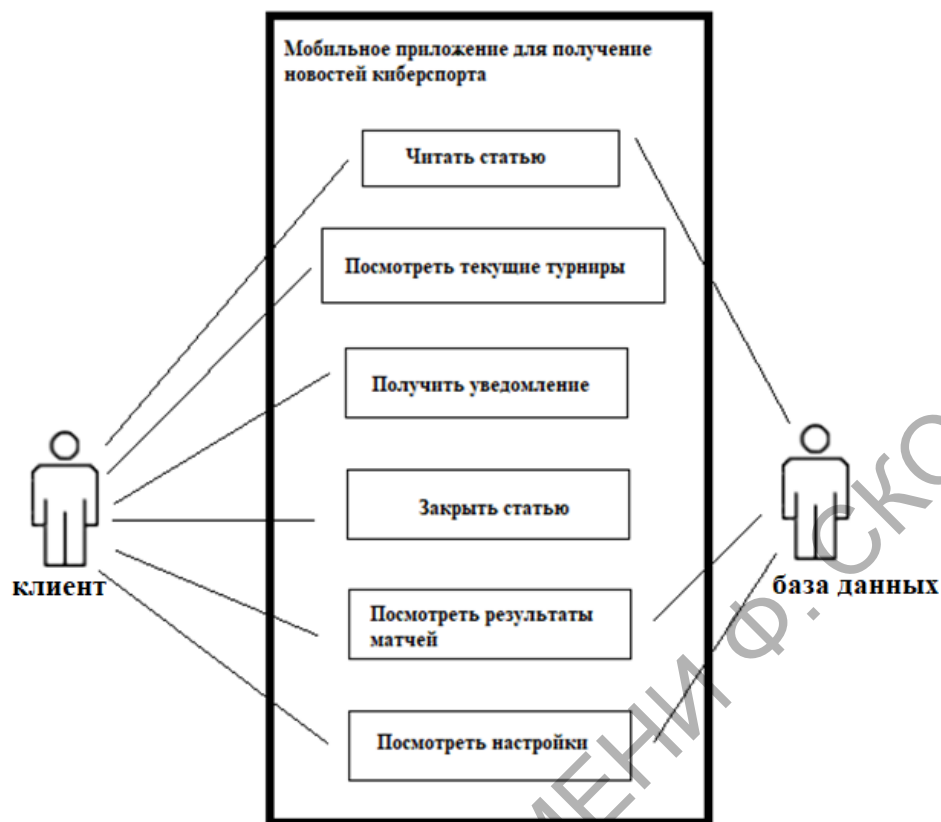


Рисунок 1 – Диаграмма вариантов использования

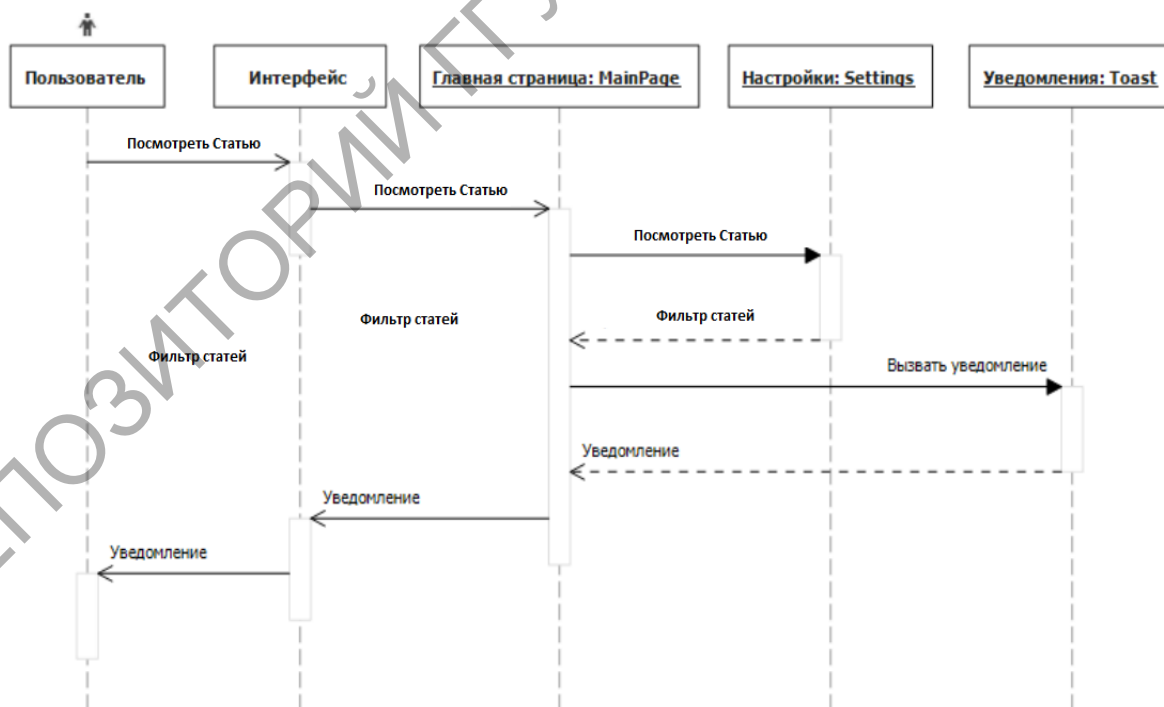


Рисунок 2 – Схема последовательности

В. В. Кутеко
(ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **Е. Е. Пугачева**, ст. преподаватель

РЕАЛИЗАЦИЯ АРХИТЕКТУРЫ ТЕСТИРУЮЩЕЙ ПРОГРАММЫ ПО ОРГАНИЗАЦИИ СИСТЕМЫ КОНТРОЛЯ ЗНАНИЙ В СРЕДНЕЙ ШКОЛЕ № 15 ГОРОДА ГОМЕЛЯ

Архитектура приложения описывает в себе четко изложенный структурный принцип, по которому создано приложение. Тестовому приложению по организации системы контроля знаний в средней школе № 15 г. Гомеля были поставлены персональные задачи, исходя из которых, была выбрана клиент-серверная архитектура.

Клиент-серверная архитектура хорошо подходит для тестового приложения, так как учащийся обращается к программе, а она в свою очередь обрабатывает данные на стороне клиента и передает их на сервер, обращаясь непосредственно к самой базе данных, которая реализована с помощью документа. Архитектура работы приложения представлена на рисунке 1.

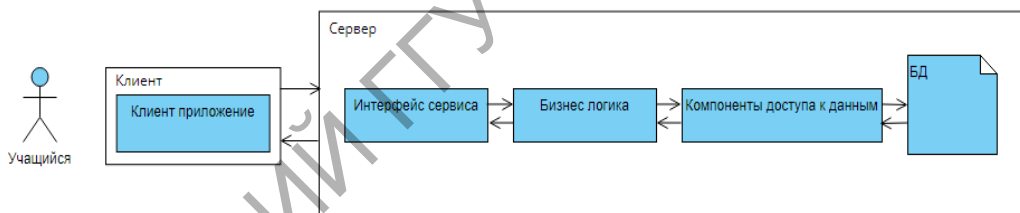


Рисунок 1 – Архитектура системы работы приложения

Архитектура работы приложения разделяет две независимые роли клиента и сервера.

Клиент отвечает за наглядный вывод данных и прямое взаимодействие с пользователем.

Сервер отвечает за интерфейс сервиса его бизнес-логику и компоненты доступа к данным. На сервере хранится информация, недоступная учащемуся, который производит операции над приложением. Эта информация представлена в виде:

- персональных данных учащегося;
- выбора тестируемой темы;
- результатов теста.

В. В. Кутеко
(ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **Е. Е. Пугачева**, ст. преподаватель

**РЕАЛИЗАЦИЯ ИНФОРМАЦИОННО-ЛОГИЧЕСКОЙ
МОДЕЛИ ДАННЫХ ТЕСТИРУЮЩЕЙ ПРОГРАММЫ
ДЛЯ ОРГАНИЗАЦИИ СИСТЕМЫ КОНТРОЛЯ ЗНАНИЙ
В СРЕДНЕЙ ШКОЛЕ № 15 ГОРОДА ГОМЕЛЯ**

Как и во всех тестовых программах «Организация системы контроля знаний в средней школе № 15 г. Гомеля» имеет свою информационно-логическую модель данных, которая реализуется путем взаимодействия персональных информационных объектов. Работа с данными производится через файлы подгружающимися автоматически при взаимодействии с тестами и главной формой авторизации. Модель данных документа охватывает аспекты создания, преобразования, хранения, поиска, передачи и отображения документов.

К основным информационным объектам документа «Soxr_rez» относятся атрибуты, представленные в таблице 1. Данные атрибуты хранят в себе упорядоченные персональные данные учащегося.

Таблица 1 – Атрибуты документа Soxr_rez

Название атрибута	Тип данных	Описание
Y_fio	text	ФИО учащегося
Y_klass	text	Класс учащегося
predm	text	Предмет
tem	text	Тема
rez	text	Результаты теста

К основным информационным объектам документа «Test» относятся атрибуты, представленные в таблице 2. Данные атрибуты хранят в себе упорядоченные выходные данные самого теста по предмету и выбранной теме.

Таблица 2 – Атрибуты документа Test

Название атрибута	Тип данных	Описание
Vopr	text	Индивидуальный вопрос
otv	text	Варианты ответа

К основным информационным объектам документа «Olim» относятся атрибуты, представленные в таблице 3. Данные атрибуты хранят в себе упорядоченные выходные данные теста по олимпиаде.

Таблица 3 – Атрибуты документа Olim

Название атрибута	Тип данных	Описание
Vopr	text	Индивидуальный вопрос
otv	text	Варианты ответа

А. О. Куценко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

АРХИТЕКТУРА ПОДСИСТЕМЫ ФОРМАЛИЗАЦИИ ВХОДЯЩИХ ДАННЫХ

В подсистеме формализации входящих данных после исследования предметной области были определены следующие роли: пользователь, приложение и источник данных. Также были выявлены следующие прецеденты: выбор типа логов, получение таблицы с обработанными логами, анализ логов, визуализация логов. На рисунке 1 представлена диаграмма вариантов использования.

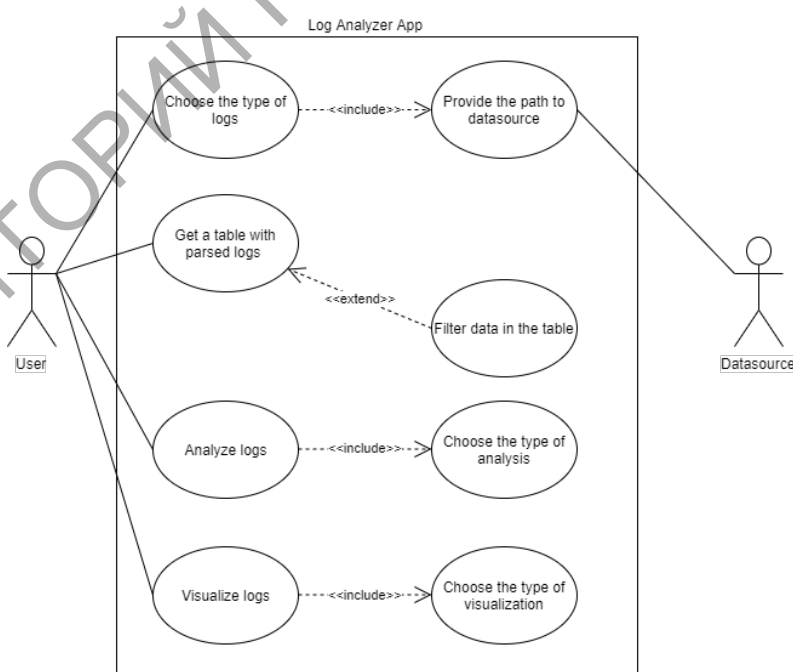


Рисунок 1 – Диаграмма вариантов использования

Для прецедента выбора типа логов представлено отношение включения с вариантом указания пути к источнику данных. Источники данных логов представлены в виде текстового файла. Пример лог-файла показан на рисунке 2.

Для прецедента получения таблицы с обработанными логами существует расширение с вариантом фильтрации данных в таблице.

Для прецедентов анализа и визуализации логов указаны отношения включения с вариантами выбора типа анализа и визуализации.

```
54.36.149.41 - - [22/Jan/2019:03:56:14 +0330] "GET /filter/27/13%20%D9%85%DA%AF%D8%A7%D9%BE%D8%8C%DA%A9%D8%B3%D9%84,27|
%DA%A9%D9%85%D8%AA%D8%B1%20%D8%A7%D8%B2%205%20%D9%85%DA%AF%D8%A7%D9%BE%D8%8C%DA%A9%D8%B3%D9%84,p53 HTTP/1.1" 200 30577 "-"
"Mozilla/5.0 (compatible; AhrefsBot/6.1; +http://ahrefs.com/robot/)" "-"
31.56.96.51 - - [22/Jan/2019:03:56:16 +0330] "GET /image/60844/productModel/200x200 HTTP/1.1" 200 5667 "https://www.zanbil.ir/m/
filter/b113" "Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.
3359.158 Mobile Safari/537.36" "-"
31.56.96.51 - - [22/Jan/2019:03:56:16 +0330] "GET /image/61474/productModel/200x200 HTTP/1.1" 200 5379 "https://www.zanbil.ir/m/
filter/b113" "Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.
3359.158 Mobile Safari/537.36" "-"
40.77.167.129 - - [22/Jan/2019:03:56:17 +0330] "GET /image/14925/productModel/100x100 HTTP/1.1" 200 1696 "-" "Mozilla/5.0
(compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)" "-"
91.99.72.15 - - [22/Jan/2019:03:56:17 +0330] "GET /product/31893/62100/
%D8%B3%D8%B4%D9%88%D8%A7%D8%B1-%D8%AE%D8%A7%D9%86%DA%AF%D8%8C-%D9%8E%D8%B1%D9%86%D8%B3%D9%84%D8%8C-%D9%85%D8%AF%D9%84-PR257AT
HTTP/1.1" 200 41483 "-" "Mozilla/5.0 (Windows NT 6.2; Win64; x64; rv:16.0)Gecko/16.0 Firefox/16.0" "-"
40.77.167.129 - - [22/Jan/2019:03:56:17 +0330] "GET /image/23488/productModel/150x150 HTTP/1.1" 200 2654 "-" "Mozilla/5.0
(compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)" "-"
40.77.167.129 - - [22/Jan/2019:03:56:18 +0330] "GET /image/45437/productModel/150x150 HTTP/1.1" 200 3688 "-" "Mozilla/5.0
(compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)" "-"
40.77.167.129 - - [22/Jan/2019:03:56:18 +0330] "GET /image/576/article/100x100 HTTP/1.1" 200 14776 "-" "Mozilla/5.0 (compatible;
bingbot/2.0; +http://www.bing.com/bingbot.htm)" "-"
66.249.66.194 - - [22/Jan/2019:03:56:18 +0330] "GET /filter/b41,b665,c150%7C%D8%A8%D8%AE%D8%A7%D8%B1%D9%BE%D8%B2,p56 HTTP/1.1"
200 34277 "-" "Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)" "-"
40.77.167.129 - - [22/Jan/2019:03:56:18 +0330] "GET /image/57710/productModel/100x100 HTTP/1.1" 200 1695 "-" "Mozilla/5.0
(compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)" "-"
207.46.13.136 - - [22/Jan/2019:03:56:18 +0330] "GET /product/10214 HTTP/1.1" 200 39677 "-" "Mozilla/5.0 (compatible; bingbot/2.
0; +http://www.bing.com/bingbot.htm)" "-"
```

Рисунок 2 – Лог-файл

А. О. Куценко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОБРАБОТКИ ДАННЫХ ЛОГОВ

При возникновении ошибок в работе программного обеспечения в самом начале изучают информацию, представленную в логах. Не имея логов, гораздо сложнее обнаружить причины ошибок, если они возникают периодически и только при определенных обстоятельствах. Запись в логи информации об ошибках и о причинах их возникновения значительно облегчает работу администраторам, DevOps-инженерам и разработчикам. По этой причине анализ логов является одним из базовых инструментов в работе IT-специалистов.

Инструменты анализа логов стали незаменимыми при устранении неполадок. С помощью инструментов анализа можно извлекать значимые данные из логов для определения основной причины любой ошибки приложения или системы, а также обнаружить тенденции и закономерности, которые могут помочь принять бизнес-решения, проводить расследования и обеспечивать безопасность.

Веб-приложение для работы с логами включает следующие функции: представление логов в табличном виде; анализ логов; визуализация логов.

Перед началом обработки лог-файлов, необходимо выбрать тип логов, так как логи, создаваемые разными приложениями, имеют разные форматы. Исходные данные представлены в виде текстового файла. Далее логи загружаются из источника, указанного пользователем, и появляется возможность получать данные логов в табличном виде, анализировать и визуализировать их.

Для разработки данного приложения был выбран язык программирования Python, а также микрофреймворк Flask и фреймворк Apache Spark. PySpark является интерфейсом для Apache Spark на Python. PySpark взаимодействует с самим Spark через специальную библиотеку Py4J. Она позволяет программам Python, которые выполняются интерпретатором, динамически обращаться к объектам Java в JVM, транслируя код Scala в JVM.

И. А. Кучерова

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **К. С. Бабич**, ст. преподаватель

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ WMI ДЛЯ СБОРА ИНФОРМАЦИИ О ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ В ОС WINDOWS

На текущий момент задача сбора информации о вычислительной системе является актуальной задачей системного администрирования. Существует огромное количество программных решений данной задачи, вот только большинство этих программ занимают достаточно много памяти и требуют установки дополнительных компонентов.

Одним из решений для ОС Windows является инструментарий управления Windows. Инструментарий весьма удобен в настройке и использовании, поскольку решение основано на стандартных механизмах Microsoft Windows.

Windows Management Instrumentation (WMI) разработан корпорацией Microsoft в рамках Web-Based Enterprise Management (WBEM), целью которой является создание стандартной технологии получения доступа к информации по управлению в среде предприятия. В инструментарии WMI используется Common Information Model (CIM) – отраслевой стандарт, служащий для представления систем, приложений, сетей, устройств и других управляемых компонентов. Модель CIM разрабатывается и поддерживается Distributed Management Task Force [1].

Получить данные из WMI можно путем написания скрипта или приложения. Создание программных продуктов, которые взаимодействуют с WMI, заключается в создании набора VBA-скриптов, или в написании MOF-файлов, или же в использовании библиотек высокоуровневых языков программирования.

При написании приложений по сбору информации с помощью языка программирования C#, используют пространство имен .NET Framework System. Management или Microsoft. Management. Infrastructure, в которых реализован базовый набор действий с инструментарием WMI.

Ниже представлен пример получения информации об операционной системе с помощью класса Win32_OperatingSystem:

- Создать новое консольное приложение C# в Visual Studio
- Добавить «System.Management» к ссылкам на проект
- Добавить пространство имен «using System.Management;» в начало файла Program.cs.

- Выполнить оставшиеся шаги внутри основного метода
- Определить запрос (то, что нужно вернуть из WMI)
`var osQuery = new SelectQuery("Win32_OperatingSystem");`
- Определить свою область действия (к какой системе нужно подключиться и путь WMI)

```
var mgmtScope =  
new ManagementScope(@"\\.\COMPUTERNAME\root\cimv2");
```

- Подключиться к WMI
`mgmtScope.Connect();`

- Определить средство поиска для запроса

```
var mgmtSrchr =  
new ManagementObjectSearcher(mgmtScope, osQuery);
```

- Вызвать метод Get поисковика и перебрать результаты
`foreach (var os in mgmtSrchr.Get())`

- Получить значение нужного свойства (версии) и сделать проверку
`var osVer = os.GetPropertyValue("Version").ToString();
if (!string.IsNullOrEmpty(osVer))`

```
Console.WriteLine("OS Version: " + osVer);
```

При корректных настройках безопасности использование WMI позволяет реализовать программы и приложения для сбора информации и отслеживанию событий, которые будут защищены от атак.

Литература

1. Сведения об инструментарии WMI [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/ru-ru/windows/win32/wmisdk/about-wmi>. – Дата доступа: 17.03.2022.

В. А. Логвинец

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ГРАФОВЫЕ БАЗЫ ДАННЫХ

Ни для кого не секрет, что работа современных интернет-сервисов и приложений так или иначе невозможна без наличия баз данных, ведь все они построены на постоянной работе с данными и нуждаются в их структурированном хранении для дальнейшего переиспользования. Существует множество разновидностей баз данных, таких как, например, реляционные, документно-ориентированные, графовые и т. д. Все они отличаются между собой по архитектуре и подходу к хранению и манипулированию данными. Вопрос выбора наиболее эффективной базы данных является одним из основополагающих в процессе разработки любого программного продукта.

Одним из решений подобной проблемы может послужить графовая база данных. Она отлично подойдёт для решения задач социальных сетей, систем рекомендаций (например, рекомендаций видео, товаров или постов на основе интересов пользователя или активности его друзей), обработки пользовательских данных, корреляции данных из различных источников (информационный след в сети).

Графовая база данных является разновидностью баз, данных с реализацией сетевой модели в виде графа и его обобщений. В общем понимании, граф – это абстрактный математический объект, представляющий собой множество вершин (точек) и рёбер (линий), то есть соединений между парами вершин. Данные, хранимые в такой базе данных, представляют собой коллекцию таких узлов (вершин) и рёбер (связей).

Узел представляет сущность (например, пользователя или организацию), а ребро – связь между двумя узлами, которые оно соединяет (например, отметка «нравится» или «являются друзьями»). Реляционная база данных может обеспечить всё, что может сделать графовая база. Однако, некоторые типы запросов можно значительно упростить именно благодаря графовой модели. Кроме того, при определенных оптимизациях некоторые запросы могут работать лучше. В отличие от реляционных, вместо сопоставления связей с таблицами и внешними ключами, графовые базы данных устанавливают связи, используя узлы, рёбра и свойства, что даёт возможность упрощённой индексации. Системами управления графовыми базами данных называют графовые СУБД. Первая такая СУБД Neo4j была реализована в 2007 году на языке программирования Java компанией «Neo Technology». Neo4j не уступает по производительности реляционным базам данных благодаря собственному формату хранения данных и может манипулировать ими по протоколам HTTP, HTTPS и BOLT (собственный протокол компании).

Таким образом, если в проекте планируется много сущностей и связей многие-ко-многим, то идеальным решением будет выбор именно графовой базы данных. Хотя они и не являются решением от всех проблем, их следует использовать для задач, которые решаются только графами.

Д. Р. Навикайте

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. А. Дробышевский**, ст. преподаватель

РАЗРАБОТКА ПРОТОТИПА САЙТА ИНТЕРНЕТ-МАГАЗИНА «FOR YOU»

Информационные технологии все больше используются в сфере торговли, существенно облегчая как жизнь покупателя, так и продавца.

На сегодняшний день сложно представить успешную торговую сеть, которая бы не имела в наличии собственный сайт, ведь интернет-магазины являются одним из эффективных способов продвижения продукции.

Продавать товары через интернет-магазины стремятся как крупные, так и мелкие производители.

Разработан интернет-магазин одежды, который позволит пользователю приобрести оригинальные и необычные вещи.

Создание интернет-магазина одежды осуществлялось с использованием технологий PHP, MySQL, HTML, CSS, JavaScript и JQuery. Для хранения информации о товарах и пользователях необходимо наличие базы данных, управление которой осуществляется посредством СУБД MySQL.

Для работы с MySQL используется скриптовый язык PHP, с помощью которого происходит обработка необходимых запросов. Например, таким образом реализуется возможность сортировки и поиска товаров, добавление и удаление товаров из корзины, её очистка, вывод товаров в определенном ценовом диапазоне, а также фильтр товаров по категориям.

Также есть разграничение прав доступа в панели администрирования интернет-магазина для администратора и пользователя. Администратор может управлять состоянием товара, изменять его описание, удалять товары и загружать новые предложения. Пользователю же доступен основной функционал, предусматривающий выбор товара, порядок заказа, который включает в себя заполнение формы с предусмотренными реквизитами.

Таким образом, разработанный интернет-магазин позволит упростить и ускорить реализацию продукции, а использование связки HTML+PHP+MySQL является мощным и удобным инструментом для его разработки.

Д. Р. Навикайте

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. А. Дробышевский**, ст. преподаватель

ИНТЕРНЕТ-МАГАЗИН ОДЕЖДЫ «FOR YOU»: ОСНОВНЫЕ ХАРАКТЕРИСТИКИ КЛИЕНТСКОЙ ЧАСТИ

Одним из основных показателей хорошего интернет-магазина является удобство его использования для клиента. Сайт должен быть прост в обращении, сочетать в себе интуитивно понятный графический интерфейс и содержать все возможности, необходимые для оптимального функционирования магазина одежды в интернете.

Разработка дизайна является важным этапом в процессе проектирования сайта, так как внешний вид интернет-магазина играет большую

роль для его посещаемости. Однако, при недостаточно разработанной и продуманной функциональной части, интернет-магазин не будет иметь эффективности. Поэтому следует уделить большое внимание двум оставшимся элементам.

В шапке сайта находится логотип интернет-магазина, навигационная панель, поисковая строка, аккаунт пользователя, а также корзина товаров. Левый блок представляет собой панель, на которой расположены инструменты, позволяющие отфильтровать продукцию по заданному критерию. В центральной части отображена предлагаемая продукция.

Реализован большой перечень функциональных возможностей. Например, есть возможность регистрации и авторизации. При нажатии на иконку профиля пользователю необходимо ввести свои личные данные для того, чтобы зарегистрироваться. Причем если данный пользователь уже существует, выведется сообщение об ошибке. У пользователя есть возможность выбора отображения товаров, которая реализуется путем нажатия на кнопку «Сортировать» и указания нужного варианта из предложенных. В интернет-магазине есть поиск товаров, обеспечивающийся вводом необходимой продукции в поисковую строку, и фильтр товаров, позволяющий отобрать продукцию определенной категории. Пользователь может добавить готовую одежду в корзину. И, в конечном итоге, заказать продукцию и оставить свой отзыв на сайте.

Таким образом, описанные характеристики клиентской части играют большую роль в предоставлении пользователю широкого спектра возможностей для приобретения продукции.

В. А. Никитюк, В. С. Сукач
(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ВЫБОР БАЗ ДАННЫХ ДЛЯ IT-ПРОЕКТОВ

Прежде всего, программисты никогда не должны самостоятельно выбирать БД для использования. В крупных корпорациях обычно есть данные и технические архитекторы, которые выполняют этот процесс.

Ни в коем случае нельзя выбирать платформу из-за комфорта или предвзятости сотрудников. Будущее решение – это система пользователей. В наше время мы должны следовать передовым практикам и

стандартам, таким как архитектура предприятия (например, TOGAF), при разработке взвешенного и более научного подхода.

Первый принцип заключается в том, чтобы следовать существующим стандартам или убедиться, что они установлены в первую очередь, и объяснить, почему. Руководящая передовая практика – свести к минимуму потребность в дополнительных ресурсах, особенно в технологии, по которой тяжело найти специалистов. Поэтому, если в вашем проекте уже есть опыт работы с Oracle, SQL Server и т.д., следует повторно использовать те же навыки работы с базами данных (RDBMS).

Это снижает затраты на технологичное оборудование и ресурсы. Поддержка новой БД требует значительных затрат и времени. Решение должно основываться на научном анализе бизнес-требований. Сначала нужно дать определение бизнес-требованиям.

Следующее решение – построить или купить готовое решение. Обычно, если бизнес-выгоды связаны с бухгалтерским учётом, закупкой запасов, управлением отношениями с клиентами (CRM), тогда вам следует обратить внимание на решения COTS, и в общем случае ERP. В этом случае ERP диктует базу данных. Например, SAP ранее позволял Oracle. Теперь он переходит на использование исключительно баз данных SAP HANA. Это очень мощная резидентная БД, но она имеет высокую цену. Microsoft Dynamics CRM использует только SQL Server.

Выбор баз данных не должен основываться на рекламе поставщика или социальных сетей. Одним из разрекламированных вариантов баз данных является NoSQL. Не стоит рассматривать решения на уровне HADOOP, если невозможно продемонстрировать 3 преимущества – объём, скорость и разнообразие. Когда ваше решение связано только с структурированными данными, то нужна реляционная или объектная база данных.

Выполнение проекта базы данных более важно, чем выбор платформы для хранения базы данных. Если вы используете Visio или не используете инструменты моделирования, вы не выполняете архитектуру данных и, следовательно, не следуете передовым методом управления данными DAMA. Неважно, насколько производительна ваша СУБД. Если дизайн или управление данными столь плохи, они не будут соответствовать требованиям пользователей.

В. С. Пашкевич
(ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **Н. А. Аксёнова**, ст. преподаватель

ОСНОВНЫЕ ПРИНЦИПЫ АНИМАЦИИ В ТРЁХМЕРНОЙ ГРАФИКЕ

3D-анимация выигрывает у классической анимации по многим пунктам: она живее, привлекательнее, ярче, эффектнее, она притягивает взгляд и интригует зрителя. Многие бренды справедливо отдают предпочтение трёхмерной анимации создавая свой имидж, рассказывая свою историю, демонстрируя свои услуги и товары, привлекая новых клиентов.

3D-анимация представляет собой формирование, размещение и манипулирование объектами в трёх плоскостях с созданием движения и эффектов. Объекты моделируют в цифровой среде с помощью инструментов и программ для скульптинга. Куда реже осязаемые объекты и люди становятся предметом сканирования и переноса в цифровую среду в качестве дальнейшей основы для 3D-анимации.

Одной из главных целей трёхмерной анимации даже при стилизации является создание и передача реалистичного движения, фактуры материалов и их взаимодействия. Для этого необходима наблюдательность, большая визуальная библиотека и исследовательская деятельность, направленная на реальные объекты окружающей среды, которые могут стать референсом или прототипом объекта анимации.

Начало любого проекта начинается с идеи и её развития, будь то книга, кино или анимация. Процесс начинается со сценария и раскадровки, разработка ключевых деталей персонажа и его наброски, разработка и скетчи фона, и техническая подготовка к анимации. В процессе задействованы многие специальности и программы, создание анимации требует большого количества деталей и умений, но вот основные этапы создания 3D-анимации.

1. Моделирование

Объектами анимации являются 3D-модели. Простая компьютерная модель – это 3D-объект из сетки, состоящей из вершин, линий и кривых, которые и придают объекту форму. Для компьютера такие модели – целостные геометрические фигуры. Объект выглядит «живым» после накладывания эффектов, текстуры и цвета.

2. Постановка и анимация

Когда 3D-персонажи готовы, их помещают в соответствующие заранее смоделированные сцены и приводят в движение, задавая каждой детали свою точку и путь движения для каждой точки и их групп. с помощью специальных программ для анимации. Анимация на начальном этапе не налажена и больше похожа на черновой вариант – движения и переходы резкие, неестественные, механические.

Первоначальный вариант анимации так выглядит, потому что сначала аниматоры создают опорные позы каждого движения. Данные позы называют ключевыми кадрами. Освещение, операторская работа (выбор угла и глубины кадра), эффекты и другие детали добавляют намного позже, когда оформляется окончательный вариант анимации, которую мы видим на экранах.

3. Рендеринг

Финальным этапом создания анимации является рендеринг. Процесс включает в себя детальную доработку анимации и получение финального изображения со всеми эффектами. Данный этап требует кропотливой работы с деталями, именно от него зависит качество финального продукта.

Данным этапом и заканчивается основной процесс создания анимации, за которым следует постпродакшн, когда добавляются спецэффекты, музыка, закадровый голос и звуковые эффекты, которые синхронизируют со всей анимацией.

В. С. Пашкевич

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Н. А. Аксёнова**, ст. преподаватель

РАЗРАБОТКА ПСЕВДОСТЕРЕОГРАФИЧЕСКОГО КОНТЕНТА ДЛЯ ПРОФОРИЕНТАЦИИ СТУДЕНТОВ ФАКУЛЬТЕТА ФИЗИКИ И ИТ

Одним из этапов решения задачи является описание схемы взаимодействия ролей участвующих в разработке и функционировании проекта. Прецеденты служат для документирования функциональных требований к программным системам. Прецедент описывает некоторый целостный фрагмент поведения системы, не вдаваясь при этом в осо-

бенности внутренней структуры субъекта. Определение прецедента содержит все свойственные ему виды поведения: основную последовательность, различные варианты стандартного поведения и различные исключительные ситуации с указанием ответной реакции на них. С точки зрения пользователя некоторые из видов поведения выглядят как ошибочные. Однако для системы ошибочная ситуация является одним из вариантов поведения, который должен быть описан и обработан.

Прецедент описывает взаимодействие программной системы с действующими лицами в виде последовательности сообщений. В понятие действующие лица входят люди, компьютерные системы и процессы. В проекте прецедентами будут являться разработчик и заказчик (рисунок 1).

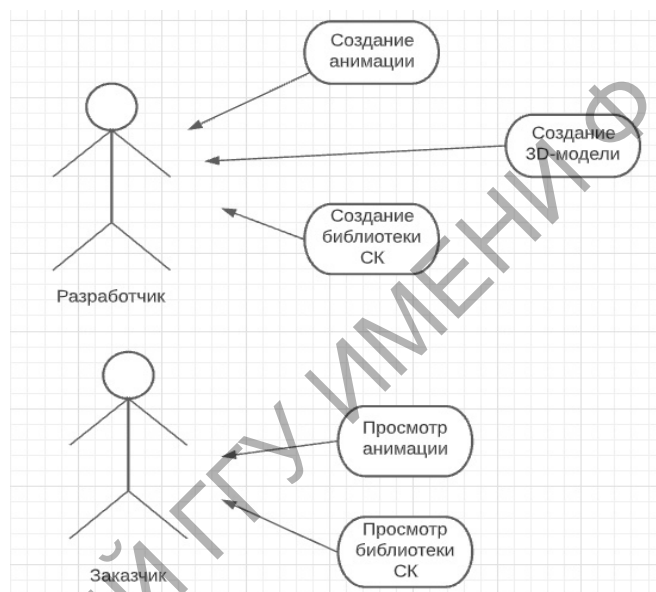


Рисунок 1 – Первоначальная диаграмма прецедентов

На диаграмме изображена схема поведения и взаимодействие между ролями. Основным блоком является библиотекой СК. Слева изображен заказчик, имеющий определенные цели и поведение в отношении данного приложения.

Главная роль заказчика – это просмотр библиотеки и проверка отображения каждой анимации.

Если вдруг какая-либо анимация будет некорректно отображаться на устройстве или не будет соответствовать требованиям заказчика – заказчик не будет удовлетворен.

Главная задача разработчика – это создание концепта анимации, разработка 3D моделей и последующая реализация анимации.

На основе проведенного анализа создадим полноценную схему (рисунок 2).

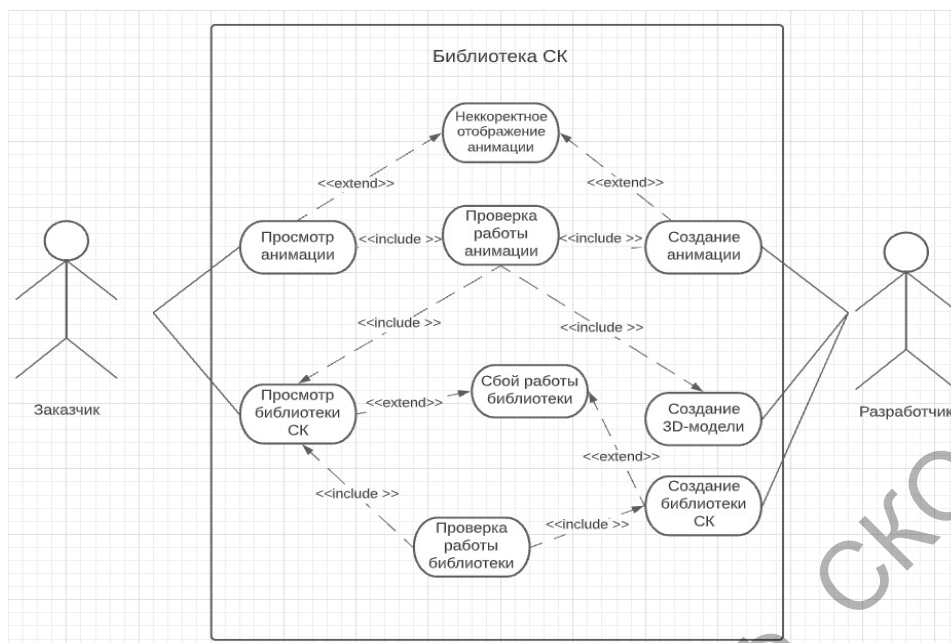


Рисунок 2 – Диаграмма прецедентов

Создание анимации включает в себя процесс анимирования 3D модели используя метод покадровой, либо программно-заданной анимации. Процесс анимации завершается рендерингом анимации, для последующего получения видеоролика. Если рендеринг будет сделан неправильно – библиотека не будет запускаться на необходимом носителе.

Т. А. Пашкевич

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. В. Грищенко**, ст. преподаватель

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ СРАВНИТЕЛЬНОГО АНАЛИЗА ОПЕРАЦИОННОЙ ЭФФЕКТИВНОСТИ НЕДВИЖИМОСТИ

В настоящее время оценка недвижимости является важным фактором во многих аспектах современного бизнеса. Например, она необходима для включения в бизнесплан. При обращении к услугам профессиональных оценщиков предполагается составление документов оценки, которыми являются заключение об оценке и отчет, являющийся неотъемлемой частью заключения, которые составляются по результатам работы. Эти документы наделяются юридической силой и могут использоваться в решении различных задач. Суд принимает данную информацию без проблем, то же касается других организаций.

Разрабатываемое веб приложение служит для решения предоставления детальной статистики и различных отчетов сравнительного анализа недвижимости.

Приложение реализован с использованием технологий и языков программирования таких как JavaScript, React, React-Query. Серверная сторона реализована на языке программирования .Net. Использование React дает возможность удобного и гибкого создание фронтенд части приложения.

JavaScript пользуется большой популярностью среди разработчиков. Это даёт возможность расширять приложение другими людьми.

React – JavaScript фреймворк, используемый для создания клиентской части веб-приложений. Благодаря компонентной архитектуре, приложения, использующие этот фреймворк легко масштабируются. React предоставляет множество функций для оптимизации, нахождения багов, анимации, сервисы и т. д. [1]. Была выбрана последняя версия фреймворка для облегчения разработчикам процесса улучшения приложения в будущем.

React-Query библиотека поможет вам извлекать, синхронизировать, обновлять и кэшировать удаленные данные, а также сократить объем кода, который вам нужно написать, предоставляя два простых хука и одну служебную функцию.

Разработка ведётся по методологии Scrum. Она позволяет грамотно распределить задачи по времени и даёт возможность другим разработчикам подключиться к созданию или улучшению приложения на любой стадии.

В процессе разработки пишется «инлайн-документация». Это подход к написанию документации, при котором во все файлы, где необходимо, добавляется описание того, что происходит внутри. Это могут быть объяснения архитектуры классов, описания функций или примеры использования.

В качестве механизма описания взаимодействия клиент-сервера используется SWAGGER. Swagger – это фреймворк для спецификации RESTful API. Его прелесть заключается в том, что он дает возможность не только интерактивно просматривать спецификацию, но и отправлять запросы – так называемый Swagger UI [2].

В результате получаем приложение, в котором клиент может добавлять, изменять и удалять данные о собственной недвижимости и проводить сравнительный анализ и бенчмарк по сравнению с другой недвижимостью на рынке.

Литература

1. Документация фреймворка React [Электронный ресурс]. – URL: <https://ru.reactjs.org/>. – Дата доступа: 23.03.2022.
2. Документация SWAGGER [Электронный ресурс]. – URL: <https://swagger.io/>. – Дата доступа: 23.03.2022.

Г. А. Пинчуков

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. В. Грищенко**, ст. преподаватель

РАЗРАБОТКА ЕДИНОГО СЕРВИСА-РЕЕСТРА РАЗМЕЩЕНИЯ БЕЗДОМНЫХ ЖИВОТНЫХ

Основной формой работы с безнадзорными владельческими и бездомными животными в особенности, с собаками – является безвозвратный отлов (то есть изъятие из городской среды без последующего возвращения животных на место отлова) и помещение отловленных животных в приюты. В большинстве из этих стран для невостребованных животных применяется усыпление. Приюты также активно действуют как центры сбора отказных, в том числе «лишних» животных у владельцев, и как центры передачи животных новым владельцам с целью предотвращения возможного пополнения популяций бездомных животных. Экспериментальная программа стерилизации с последующим выпуском собак в места поимки, по данным на март 2011 года, планируется к введению в стране, из европейских стран проводится лишь в Греции Одним из последствий экспериментов со стерилизацией бродячих собак в городах стало появление в начале 2010-х годов догхантеров. Свободное обитание собак на улицах городов считается неприемлемым Национальной ассоциацией по контролю над животными США. Согласно заключению этой организации, при свободном обитании собаки подвергаются эпидемическому риску (заражению от других животных), они могут нападать на домашний скот или убивать других животных, могут являться причиной жестоких проявлений со стороны недовольных хозяев других животных, могут отравиться и умереть в муках после приёма пищи, найденной среди мусора, становятся причиной ДТП и других происшествий. Ассоциация констатирует необходимость отлова и усыпления невостребованных животных в приютах. В 2013 году в США было усыплено более 2 700 000 собак и кошек,

попавших в приюты, но не нашедших себе новых хозяев. Вся наша площадка по контролю бездомных животных будет поддерживаться сервером. Сервер – мозг для нашего приложения. Он будет контролировать всю информацию по животным и представлять её для человека в том виде, в котором она будет удобна для анализа и управления.

Сервер будет реализован с использованием технологий и языков программирования таких как JavaScript, Node.js, Express, Angular 13. Это обеспечит возможность расширять возможности сервера, не нарушая работы его существующей функциональности. Использование Angular даст возможность удобного управления сервером и компонентами системы.

Весь стек технологий использует язык JavaScript. Он пользуется большой популярностью среди разработчиков. Это даёт возможность расширять приложение другими людьми.

Node.js – среда выполнения языка JavaScript на локальной машине. Она как правило используется для создания компьютерных приложений, серверов и т.д. Для создания требуемого сервера был выбран фреймворк Express, так как он прост в изучении и использовании.

Express – JavaScript фреймворк, используемый для написания серверной части веб-приложений. Позволяет рендерить шаблон страницы на сервере или создавать API, отправляющее данные клиенту в формате JSON.

Angular – JavaScript фреймворк, используемый для создания клиентской части веб-приложений. Благодаря компонентной архитектуре, приложения, использующие этот фреймворк легко масштабируются. Angular предоставляет множество функций для оптимизации, нахождения багов, анимации, сервисы и т. д. [1]. Была выбрана последняя версия фреймворка для облегчения разработчикам процесса улучшения приложения в будущем.

В качестве базы данных выбрана PostgreSQL. Она имеет множество встроенных функций и, благодаря большому сообществу разработчиков, позволяет легко найти их описание. PostgreSQL предоставляет собственную систему управления и поддерживается большинства другими системами управления [2].

В результате получаем архитектуру приложения, в которой человек отправляет запрос на сервер с некоторыми данными о своём состоянии, сервер записывает полученную информацию в базу данных и отправляет сообщение об успешной операции или ошибке в приложение.

Предполагаемый результат – рабочий прототип приложения, использующего перечисленные ранее технологии. Главной целью разработки является создания масштабируемого приложения для управления и сбора информации о бездомных животных.

Литература

1. Документация фреймворка Angular [Электронный ресурс]. — URL: <https://angular.io/>. — Дата доступа: 12.03.2022.
2. Документация PostgreSQL [Электронный ресурс]. — URL: <https://www.postgresql.org/docs/>. — Дата доступа: 12.03.2022.

М. А. Раздуев

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

СВЕРХБОЛЬШИЕ БАЗЫ ДАННЫХ

Сверхбольшая база данных (VLDB) – это база данных, которая занимает максимально возможные объёмы БД на устройстве физического хранения. VLDB состоит из очень большого количества записей, строк и записей базы данных.

VLDB похожа на стандартную базу данных, но содержит очень большой объем данных. Таким образом, для его работы требуется особое управление, техническое обслуживание и технологии.

Нет какого-то определенного количества данных, которые мы могли бы привести. Например, нельзя сказать, что любая база данных с объемом данных более 1 ТБ считается VLDB. Этот абсолютный объем данных менялся с течением времени, поскольку компьютерные методы обработки, хранения и резервного копирования стали лучше справляться с большими объемами данных. Тем не менее, проблемы с VLDB могут начать появляться при приближении к 1 ТБ и, более чем вероятно, появятся при превышении 30 ТБ или около того.

Одни из более известных систем сверхбольших баз данных являются BaBar, EOS/DIS, SkyServer.

Как было сказано ранее, одной из самых больших научных баз данных является база данных проекта BaBar. Эксперимент BABAR в SLAC был одним из первых экспериментов HER, в котором использовалась объектно-ориентированная система управления базами данных

для основного хранилища событий. Цель этого эксперимента заключалась в изучении нарушений CP-симметрии при распаде B-мезонов. Ежедневно поставляется около 500 ГБайт информации.

Стоит также рассмотреть сверхбольшую базу данных проекта EOS/DIS (Earth Observation System/Data Information System), разрабатываемого агентством NASA в США. Эта система дает возможности по управлению данными по науке о Земле из самых разнообразных источников, таких как спутники, самолеты, полевые измерения. EOS/DIS принимает, обрабатывает, архивирует и распространяет данные с большого количества источников, которые были упомянуты ранее. На сентябрь 2012 года было известно, что EOS/DIS содержит около 10 ПБ данных в своей базе данных, и с каждым днем прибавляется около 8,5 ТБ.

Следующим примером системы является система SkyServer проекта SDSS (Sloan Digital Sky Survey). SkyServer позволяет исследовать звезды и галактики, которые изучают профессиональные астрономы. Целью SDSS является создать качественную трехмерную карту Вселенной. Для съемок используется 2,5-метровый телескоп, ПЗС-камера и различное программное обеспечение для хранения, обработки и анализа данных.

А. В. Сапанович

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ИСПОЛЬЗОВАНИЕ REST В ASP.NET ВЕБ-ПРИЛОЖЕНИЯХ

API или интерфейс прикладного программирования – это набор правил, которые определяют, как приложения или устройства могут подключаться и взаимодействовать друг с другом. REST API – это API, который соответствует принципам проектирования REST или архитектурному стилю репрезентативной передачи состояния. По этой причине REST API иногда называют RESTful API.

REST, впервые определенный в 2000 году Роем Филдингом в его докторской диссертации, обеспечивает относительно высокий уровень гибкости и свободы для разработчиков. Эта гибкость – лишь одна из причин, по которой REST API стали распространенным методом соединения компонентов и приложений в архитектуре микросервисов.

Некоторые API, такие как SOAP или XML-RPC, налагают на разработчиков строгие рамки. Но REST API можно разрабатывать практически на любом языке программирования и поддерживать различные форматы данных. Единственное требование состоит в том, чтобы они соответствовали следующим шести принципам проектирования REST, также известным как архитектурные ограничения:

- единый интерфейс;
- клиент-серверная архитектура;
- отсутствие состояний;
- кэшируемость;
- многоуровневая архитектура;
- код по запросу (необязательно).

В ASP.NETRESTful приложения создаются на базе стандартного шаблона Web-API. API-интерфейсы REST взаимодействуют через HTTP-запросы для выполнения стандартных функций базы данных, таких как создание, чтение, обновление и удаление записей (также известных как CRUD) в ресурсе. Например, REST API будет использовать запрос GET для извлечения записи, запрос POST для ее создания, запрос PUT для обновления записи и запрос DELETE для ее удаления. Все методы HTTP могут использоваться в вызовах API. Хорошо спроектированный REST API похож на веб-сайт, работающий в веб-браузере со встроенными функциями HTTP.

А. В. Сапанович

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

РАЗРАБОТКА ASP.NET-ПРИЛОЖЕНИЯ С ЗАЛОЖЕННЫМИ ОШИБКАМИ

Развитие информационных технологий привело к увеличению важности такого этапа разработки приложений как тестирование. Это очень важный и необходимый инструмент, который можно и нужно использовать для минимизации количества ошибок в приложении.

По этой причине было разработано приложения с заложенными ошибками для отбора кандидатов в тестировщики. Данное приложение призвано облегчить и улучшить отбор потенциальных сотрудников в IT компании путем автоматизации и унификации отборочных заданий.

Приложение представляет собой клиент-серверное web-приложение, представляющее собой систему обратной связи для сотрудников компании, в функционал которого входит проведение опросов сотрудников на различные темы, касающиеся условий труда, создание и обработка заявок на закупку расходных материалов, техники, рассылка уведомлений на электронную почту и мессенджеры и т. д.

Приложение разработано на фреймворке ASP.NET Core на языке C#, и состоит из back-end и front-end частей, отвечающих соответственно за обработку операций на сервере и их визуализацию.

Серверная часть предоставляет REST API для работы с клиентской частью, а также отвечает за работу с базой данных. Можно выделить два основных компонента: слой доступа к данным и слой бизнес-логики. Первый используется для предоставления доступа и обработки данных, и, главным образом, отвечает за подключение к БД и обработки CRUD-операций на низком уровне. Второй, в свою очередь, занят маршрутизацией HTTP-запросов front-end части приложения, валидацией и обработкой полученных данных.

Разделение данных приложения и управляющей логики позволяет модифицировать эти компонента независимо друг от друга, что облегчает разработку, обслуживание и поддержку проекта. В частности, за счет такого разделения увеличивается возможность повторного использования кода.

Д. А. Хвесюк

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **Н. А. Аксёнова**, ст. преподаватель

РАЗРАБОТКА ДИАГРАММЫ ПРЕЦЕДЕНТОВ ДЛЯ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ АДАПТИВНОГО ПРОЦЕССА ИЗУЧЕНИЯ ТЕХНОЛОГИИ MACHINE LEARNING

Диаграмма вариантов использования в UML – диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Функциональные возможности проекта: изучение терминов, курсов и других учебных материалов, отслеживание прогресса, проверка знаний посредством теоретических и практических заданий.

На данный момент в системе имеются два вида акторов: пользователь и администратор. Пользователю доступен режим чтения данных. Администратор в свою очередь обеспечивает создание, обновление и удаление контента. Диаграмма представлена на рисунке 1.

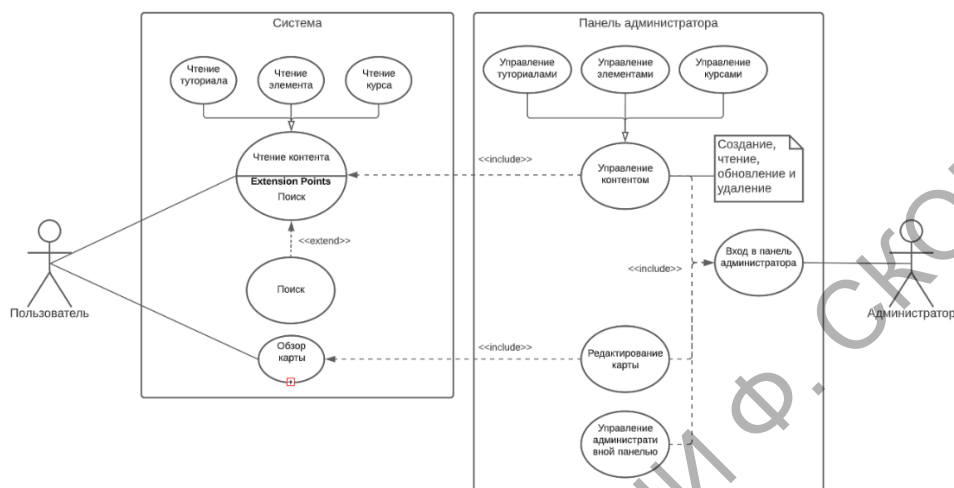


Рисунок 1 – Диаграмма прецедентов

Система разработана с использованием фреймворка Django. За панель администратии отвечает стандартный модуль фреймворка Django – Django-Admin. Он «из коробки» предоставляет доступ для создания, редактирования и удаления данных.

В будущем в системе будет представлен еще один актор: зарегистрированный пользователь, который будет обладать большим спектром функциональных возможностей: создание заметок, контроль своего образовательного процесса, получение рекомендаций на основе машинного обучения.

Г. П. Хоронеко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

РАЗРАБОТКА ПРЕДСТАВИТЕЛЬСКОГО САЙТА ДЛЯ ОАО «БЭРН»

Целью разрабатываемого сайта-визитки является простое, быстрое и удобное донесение информации об организации до конечного поль-

зователя. Создание возможности быстрой обратной связи за счет электронных обращений, а также возможности откликнуться на вакансию о работе на сайте организации, не используя сторонние ресурсы.

C# это мощный и одновременно гибкий язык программирования. Последнюю пару лет C# играл важную роль в производстве устойчивых к сбоям продуктов – от настольных приложений до Web-сервисов, от высокоуровневых решений в автоматизации бизнес-процессов до программ системного уровня и от однопользовательских продуктов до корпоративных решений в сетевых распределенных средах.

Язык C# сертифицирован Международной организацией по стандартизации (ISO). Таким образом, эта разработка Microsoft имеет статус международных стандартов. Также, язык C# признан в качестве стандарта Европейской ассоциацией производителей компьютеров (ЕСМА).

Представительский сайт для компании ОАО «БЭРН» предоставляет информацию об организации, истории, руководстве предприятия, возможность отклика на активные вакансии, а также возможность оставить обращение. На сайте можно увидеть карту, где находятся филиалы предприятия.

Для разрабатываемого сайта-визитки были описаны основные сценарии пользования, составлены UML-диаграммы прецедентов, приведена информационно-логическая модель данных и архитектура проекта.

Полученный сайт-визитка полностью выполняет свой функционал и является отличным решением для малого и среднего бизнеса.

Г. П. Хоронеко

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ОСНОВНЫЕ МЕХАНИЗМЫ САЙТА-ВИЗИТКИ ОАО «БЭРН»

Целью разрабатываемого сайта-визитки является простое, быстрое и удобное донесение информации об организации до конечного пользователя. Создание возможности быстрой обратной связи за счет электронных обращений, а также возможности откликнуться на вакансию о работе на сайте организации, не используя сторонние ресурсы.

Основная информация хранится в базе данных.

Входными данными на сайте-визитке ОАО «БЭРН» представлены: ФИО, телефон заявителя для связи, место работы (учебы), почтовый адрес места жительства, адрес электронной почты заявителя, тема обращения, а также текст обращения, специальность, квалификация, опыт работы, период работы. Данная информация вносится в соответствующие формы для заполнения на обращение, либо отклик на вакансию.

Выходной информацией являются уведомления о регистрации откликов в базе данных.

На сайте-визитке имеется возможность входа администратора и добавления новостей, а также редактирования вакансий.

Полученный сайт-визитка полностью выполняет свой функционал и является отличным решением для малого и среднего бизнеса.

Содержание

Для перехода к тексту материалов перейдите по гиперссылке
на соответствующем докладе

Секция 1 «Новые материалы и технологии»

Маевский А. А., Данильченко К. Д. Влияние параметров золь-гель процесса на формирование сегнетоэлектрических золь-гель плёнок.....	6
Москвичёв М. И. Разработка методики получения микросферических материалов с применением золь-гель метода и распылительного пиролиза.....	8
Яночкин П. С., Данильченко К. Д., Маевский А. А. Оценка и анализ электрофизических свойств золь-гель плёнок на основе SiO ₂	11

Секция 2 «Моделирование физических процессов»

Павленко А. В. Приближённое решение двумерного уравнения Логанова–Тавхелидзе в случае потенциала Гаусса.....	15
---	----

Секция 3 «Автоматизация исследований»

Ramanau I. A. Development a mobile application for educational content management.....	20
Zhaldakova A. A. Content management system.....	22
Zhaldakova A. A. The main goals of business card sites, their main advantages.....	23
Бенько Д. И. Основные механизмы подсистемы учёта закупок для типовой конфигурации «1С: Предприятие».....	24
Бенько Д. И. Разработка подсистемы учёта закупок для типовой конфигурации «1С: Предприятие».....	25
Бобкова М. В., Троцкая Д. Д. Использование нереляционных баз данных	26
Богданович Д. В. Изучение сигналов интерфейса IrDA с помощью логического анализатора.....	27

Графов А. Р. Разработка наборов тестов для проверки функционирования сайтов.....	29
Графов А. Р. Применение технологии Selenium WebDriver при написании автоматизированных тестов.....	30
Гулевич М. И. Разработка мобильного приложения поиска оптимального режима для фитнес-тренинга.....	31
Дашук А. А. Разработка подсистемы учета услуг транспортной организации.....	32
Дашук А. А. Автоматизация учета услуг транспортной организации	33
Дейниченко В. В. Тестирование веб-приложения.....	35
Дейниченко В. В. Разработка фреймворка для тестирования веб-приложения.....	36
Дробышевский А. С. Создание и настройка Linux агента для CI/CD системы TeamCity.....	37
Зубрицкий А. А. Применение средств языка Python для обработки данных, полученных СВЧ резонаторным методом.....	40
Киёк И. А. Разработка веб-приложения новостей CyberNews.....	43
Киёк И. А. Архитектура веб-приложения новостей CyberNews.....	44
Кутеко В. В. Реализация архитектуры тестирующей программы по организации системы контроля знаний в средней школе № 15 города Гомеля	46
Кутеко В. В. Реализация информационно-логической модели данных тестирующей программы для организации системы контроля знаний в средней школе № 15 города Гомеля.....	47
Куценко А. О. Архитектура подсистемы формализации входящих данных	48
Куценко А. О. Разработка приложения для обработки данных логов.	49
Кучерова И. А. Использование технологии WMI для сбора информации о вычислительной системе в ОС Windows.....	50
Логвинец В. А. Графовые базы данных.....	52
Навикайте Д. Р. Разработка прототипа сайта интернет-магазина «For You»	53
Навикайте Д. Р. Интернет-магазин одежды «For You»: основные характеристики клиентской части.....	54
Никитюк В. А., Сукач В. С. Выбор баз данных для IT-проектов.....	55
Пашкевич В. С. Основные принципы анимации в трёхмерной графике	57
Пашкевич В. С. Разработка псевдостереографического контента для профорientации студентов факультета физики и ИТ.....	58
Пашкевич Т. А. Разработка веб-приложения для сравнительного анализа операционной эффективности недвижимости.....	60
Пинчуков Г. А. Разработка единого сервиса-реестра размещения бездомных животных.....	62
Раздужев М. А. Сверхбольшие базы данных.....	64

Сапанович А. В. Использование REST в ASP.NET веб-приложениях	65
Сапанович А. В. Разработка ASP.NET-приложения с заложенными ошибками.....	66
Хвесюк Д. А. Разработка диаграммы прецедентов для веб-приложения для адаптивного процесса изучения технологии Machine Learning	67
Хоронеко Г. П. Разработка представительского сайта для ОАО «БЭРН»	68
Хоронеко Г. П. Основные механизмы сайта-визитки ОАО «БЭРН»...	69

Научное электронное издание

АКТУАЛЬНЫЕ ВОПРОСЫ ФИЗИКИ И ТЕХНИКИ

XI Республиканская научная конференция
студентов, магистрантов и аспирантов,
посвященная 100-летию со дня рождения
академика Белого Владимира Алексеевича

(Гомель, 21 апреля 2022 года)

Сборник материалов

В двух частях

Часть 2

Подписано к использованию 09.06.2022.

Объём издания 2,5 МБ.

Издатель и полиграфическое исполнение:
учреждение образования

«Гомельский государственный университет имени Франциска Скорины».

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 3/1452 от 17.04.2017.

Специальное разрешение (лицензия) № 02330 / 450 от 18.12.2013.

ул. Советская, 104, 246028, Гомель.

<http://conference.gsu.by>