

Школа ИШИТР

Направление подготовки Информационные системы и технологии

Отделение школы (НОЦ) Отделение информационных технологий

БАКАЛАВРСКАЯ РАБОТА

Тема работы
Разработка фреймворка автоматизации тестирования программного обеспечения

УДК 004.415.2:004.415.53

Студент

Группа	ФИО	Подпись	Дата
8И8Б	Савченко Константин Дмитриевич		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Мирошниченко Евгений Александрович	Кандидат технических наук		

КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Меньшикова Екатерина Валентиновна	Кандидат философских наук		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна	-		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапка Ирина Валериевна	Кандидат технических наук		

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа ИШИТР

Направление подготовки (специальность) Информационные системы и технологии

Отделение школы (НОЦ) Отделение информационных технологий

УТВЕРЖДАЮ:

Руководитель ООП

_____ Цапко И. В.
 (Подпись) (Дата) (Ф.И.О.)

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

В форме:

Бакалаврской работы

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8И8Б	Савченко Константин Дмитриевич

Тема работы:

Разработка фреймворка автоматизации тестирования программного обеспечения

Утверждена приказом директора (дата, номер)	
---	--

Срок сдачи студентом выполненной работы:	
--	--

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

<p>Исходные данные к работе</p> <p><i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Требования от заказчика на автоматизацию тестирования программного обеспечения</p>
---	---

<p>Перечень подлежащих исследованию, проектированию и разработке вопросов <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<p>1) Анализ области автоматизации тестирования программного обеспечения; 2) Анализ тестируемого приложения; 3) Выбор инструмента для автоматизации тестирования пользовательского интерфейса; 4) Разработка архитектуры автоматизированных тестов; 5) Реализация набора автоматизированных тестов пользовательского интерфейса для регрессионного тестирования тестируемого приложения;</p>
<p>Перечень графического материала <i>(с точным указанием обязательных чертежей)</i></p>	<p>-</p>
<p>Консультанты по разделам выпускной квалификационной работы <i>(с указанием разделов)</i></p>	
<p>Раздел</p>	<p>Консультант</p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Меньшикова Екатерина Валентиновна, доцент</p>
<p>Социальная ответственность</p>	<p>Мезенцева Ирина Леонидовна, старший преподаватель</p>
<p>Названия разделов, которые должны быть написаны на русском и иностранном языках:</p>	
<p>-</p>	

<p>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</p>	
--	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Мирошниченко Евгений Александрович	Кандидат технических наук		24.01.2022

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8И8Б	Савченко Константин Дмитриевич		

Министерство науки и высшего образования Российской Федерации
 федеральное государственное автономное
 образовательное учреждение высшего образования
 «Национальный исследовательский Томский политехнический университет» (ТПУ)

Школа ИШИТР

Направление подготовки (специальность) Информационные системы и технологии

Уровень образования Бакалавр

Отделение школы (НОЦ) Отделение информационных технологий

Период выполнения весенний семестр 2021/2022 учебного года

Форма представления работы:

Бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН выполнения выпускной квалификационной работы

Срок сдачи студентом выполненной работы:	10.06.2022
--	------------

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
	Формулировка задач разработки	
	Описание предметной области	
	Анализ методов решения задачи	
	Разработка архитектуры решения	
	Реализация сценариев автоматизированных тестов	
	Анализ результатов	
	Социальная ответственность	
	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	

СОСТАВИЛ:

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Мирошниченко Евгений Александрович	Кандидат технических наук		

СОГЛАСОВАНО:

Руководитель ООП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Цапко Ирина Валериевна	Кандидат технических наук		

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ООП

Код компетенции	Наименование компетенции
Универсальные компетенции	
УК(У)-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач
УК(У)-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений
УК(У)-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде
УК(У)-4	Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(-ых) языке(-ах)
УК(У)-5	Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах
УК(У)-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни
УК(У)-7	Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности
УК(У)-8	Способен создавать и поддерживать в повседневной жизни и в профессиональной деятельности безопасные условия жизнедеятельности для сохранения природной среды, обеспечения устойчивого развития общества, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных конфликтов
УК(У)-9	Способен проявлять предприимчивость в практической деятельности, в т.ч. в рамках разработки коммерчески перспективного продукта на основе научно-технической идеи
УК(У)-10	Способен принимать обоснованные экономические решения в различных областях жизнедеятельности
УК(У)-11	Способен формировать нетерпимое отношение к коррупционному поведению
Общепрофессиональные компетенции	
ОПК(У)-1	Способен применять естественнонаучные и общепрофессиональные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности
ОПК(У)-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
ОПК(У)-3	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-

Код компетенции	Наименование компетенции
Универсальные компетенции	
	коммуникационных технологий и с учетом основных требований информационной безопасности
ОПК(У)-4	Способен участвовать в разработке технической документации, связанной с профессиональной деятельностью с использованием стандартов, норм и правил
ОПК(У)-5	Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем
ОПК(У)-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
ОПК(У)-7	Способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем
ОПК(У)-8	Способен применять математические модели, методы и средства проектирования информационных и автоматизированных систем
Профессиональные компетенции	
ПК(У)-1	Способен разрабатывать требования и проектировать программное обеспечение
ПК(У)-2	Способен выполнять работы, связанные со сбором, обработкой и подготовкой картографической информации
ПК(У)-3	Способен выполнять анализ и интерпретацию данных ДЗЗ
ПК(У)-4	Способен выполнять работы и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы
ПК(У)-5	Способен разрабатывать графический дизайн интерфейса
ПК(У)-6	Способен разрабатывать базы данных ИС
ПК(У)-7	Способность обеспечивать безопасность информации в автоматизированных системах

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа		ФИО	
8И8Б		Савченко Константин Дмитриевич	
Школа	ИШИТР	Отделение (НОЦ)	Отделение информационных технологий
Уровень образования	Бакалавриат	Направление/специальность	09.03.02 Информационные системы и технологии

Тема ВКР:

Разработка фреймворка автоматизации тестирования программного обеспечения	
Исходные данные к разделу «Социальная ответственность»:	
<p>Введение</p> <p>– Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика) и области его применения.</p> <p>– Описание рабочей зоны (рабочего места) при разработке проектного решения/при эксплуатации</p>	<p><i>Объект исследования:</i> процесс тестирования ПО в IT-компании</p> <p><i>Область применения:</i> разработка программных продуктов</p> <p><i>Рабочая зона:</i> офис</p> <p><i>Размеры помещения (климатическая зона*)</i> 6*8 м.</p> <p><i>Количество и наименование оборудования рабочей зоны ПЭВМ.</i></p> <p><i>Рабочие процессы, связанные с объектом исследования, осуществляющиеся в рабочей зоне:</i> Разработка и отладка автотестов, анализ их выполнения. Выявление ошибок в тестируемом программном обеспечении. Доработка фреймворка написания автотестов.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>1. Правовые и организационные вопросы обеспечения безопасности при разработке проектного решения</p> <p>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</p> <p>– организационные мероприятия при компоновке рабочей зоны.</p>	<p>ГОСТ 12.2.032-78 Рабочее место при выполнении работ сидя;</p> <p>ГОСТ 21889-76 Система "Человек-машина". Кресло человека-оператора;</p> <p>ТК РФ Статья 173. Гарантии и компенсации работникам, совмещающим работу с получением высшего образования по программам бакалавриата, программам специалитета или программам магистратуры, и работникам, поступающим на обучение по указанным образовательным программам;</p>
<p>2. Производственная безопасность при разработке проектного решения</p> <p>– Анализ выявленных вредных и опасных производственных факторов</p>	<p>Вредные производственные факторы:</p> <ol style="list-style-type: none"> 1. Зрительное напряжение; 2. Отклонение показателей микроклимата в закрытом помещении; 3. Недостаточная освещенность;

	<p>4. Нервно-психические перегрузки, монотонность трудового процесса.</p> <p>Опасные факторы:</p> <p>1. Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.</p> <p>Требуемые средства коллективной и индивидуальной защиты от выявленных факторов:</p> <p>1. Устройства автоматического отключения, предохранительные устройства электросети.</p>
<p>3. Экологическая безопасность при разработке проектного решения</p>	<p>Воздействие на литосферу: отходы при поломке предметов вычислительной техники и оргтехники, бытовой мусор</p>
<p>4. Безопасность в чрезвычайных ситуациях при разработке проектного решения</p>	<p>Возможные ЧС: Пожар; Обрушение здания; Авария на коммунальных системах; Наиболее типичная ЧС возгорание вычислительной техники из-за перегрузки</p>
<p>Дата выдачи задания для раздела по линейному графику</p>	<p>01.03.2022</p>

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Мезенцева Ирина Леонидовна	-		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8И8Б	Савченко Константин Дмитриевич		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8И8Б	Савченко Константин Дмитриевич

Школа	ИШИТР	Отделение школы (НОЦ)	Отделение информационных технологий
Уровень образования	Бакалавриат	Направление/специальность	09.03.02 Информационные системы и технологии

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i>	Бюджет – 170 337 руб. Затраты на заработную плату – 84 993руб.
2. <i>Нормы и нормативы расходования ресурсов</i>	Тариф на электроэнергию 3,85 кВт/ч
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Налог во внебюджетные фонды 30.0 Районный коэффициент – 1,3 Накладные расходы – 16%

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>	Оценка коммерческого потенциала: 1. Потенциальные потребители результатов исследования. 2. Анализ конкурентных технических решений 3. SWOT-анализ
2. <i>Планирование и формирование бюджета научных исследований</i>	Расчет бюджета научных исследований.
3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	Определена ресурсная, финансовая, бюджетная эффективность исследования посредством расчета интегрального финансового показателя, интегрального показателя ресурсоэффективности и эффективности.

Перечень графического материала (с точным указанием обязательных чертежей):

1. <i>Оценка конкурентоспособности технических решений</i>
2. <i>Матрица SWOT</i>
3. <i>Альтернативы проведения НИ</i>
4. <i>График проведения и бюджет НИ</i>
5. <i>Оценка ресурсной, финансовой и экономической эффективности НИ</i>

Дата выдачи задания для раздела по линейному графику	25.04.2022
---	------------

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Меньшикова Екатерина Валентиновна	К.ф.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8И8Б	Савченко Константин Дмитриевич		

РЕФЕРАТ

Выпускная квалификационная работа 90 с., 13 рис., 29 табл., 22 источников, 4 прил.

Ключевые слова: автоматизация тестирования, тестирование пользовательского интерфейса, тестирование веб-приложений, Selenium, архитектура автоматизированных тестов.

Объектом исследования является (ются) процесс автоматизации тестирования пользовательского интерфейса веб-приложения.

Цель работы – автоматизация тестирования пользовательского интерфейса веб-приложения Интернет-магазин САО “ВСК”.

В процессе исследования проводились анализ тестируемого приложения на предмет выявления требований к автоматизированным тестам, анализ инструментов автоматизации тестирования пользовательского интерфейса, изучение подходов к написанию автоматизированных тестов пользовательского интерфейса, разработка архитектуры для написания автоматизированных тестов, реализация тестов для регрессионного тестирования веб-приложения Интернет-магазин САО “ВСК”.

В результате исследования был спроектирован и разработан проект автоматизации тестирования пользовательского интерфейса, созданы наборы автоматизированных тестов.

Степень внедрения: реализованные автоматизированные тесты используются для регрессионного тестирования при поставках новых версий системы Интернет-магазин “ВСК”.

Область применения: долгосрочные проекты по разработке программного обеспечения.

Экономическая эффективность/значимость работы сокращение времени на регрессионное тестирование, сокращение расходов на ручное тестирование.

В будущем планируется расширение реализованного проекта для автоматизации тестирования в новых системах предприятия-заказчика.

Содержание

Введение.....	14
Автоматизация тестирования пользовательского интерфейса	16
Проблематика области автоматизации тестирования	16
Пирамида тестирования	17
Анализ предметной области	19
Описание тестируемого приложения.....	19
Анализ тестовых сценариев	20
Используемые средства автоматизации тестирования	21
Шаблоны проектирования тестов пользовательского интерфейса	24
Реализация проекта автоматизированных тестов.....	26
Применение паттерна Page Object. Уровень страницы. Element Object	26
Операции уровня бизнес-логики	29
Алгоритм выполнения тестового метода	31
Работа с веб-драйвером браузеров	32
Взаимодействие с API тестируемой системы	34
Модель данных теста.....	36
Логирование.....	38
Заключение	40
Социальная ответственность	43
Введение.....	43
Правовые и организационные вопросы обеспечения безопасности	44
Производственная безопасность	46

Экологическая безопасность.....	51
Безопасность в чрезвычайных ситуациях.....	52
Вывод.....	55
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	56
Введение.....	56
1.1 Потенциальные потребители результатов исследования.....	57
1.2 Анализ конкурентных технических решений.....	57
1.4 SWOT анализ.....	59
2 Определение возможных альтернатив проведения научных исследований.....	62
3.1 Структура работ в рамках научного исследования.....	63
3.2 Определение трудоемкости выполнения работ.....	63
3.3 Разработка графика проведения научного исследования.....	64
4.1 Расчёт материальных затрат НИИ.....	68
4.2 Расчёт затрат на специальное оборудование для научных (экспериментальных) работ.....	69
4.3 Расчёт основной заработной платы исполнителей системы....	70
4.4 Дополнительная заработная плата исполнителей темы.....	73
4.5 Отчисления во внебюджетные фонды (страховые отчисления)	73
4.6 Расчет накладных расходов.....	74
4.5 Формирование бюджета затрат научно-исследовательского проекта.....	75

5 Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования	76
Вывод.....	78
Список использованных источников	80
Приложение 1	84
Приложение 2	85
Приложение 3	86
Приложение 4	89

Введение

Обеспечение качества (Quality Assurance (QA)) — представляет собой совокупность мероприятий, охватывающих абсолютно все этапы разработки, выпуска и эксплуатации программного обеспечения. Это активности на всех этапах жизненного цикла ПО, которые предпринимаются для обеспечения заявленного требованиями уровня качества выпускаемого продукта. Контроль качества (Quality Control (QC)) — это часть комплекса QA в процессе разработки ПО, которая отвечает за анализ результатов тестирования, поиск ошибок и их устранение. Тестирование программного обеспечения (Software Testing) — одна из техник контроля качества, включающая в себя активности по планированию тестовых действий, дизайну тестов, их выполнению и анализу полученных данных т.е. уже непосредственно процесс проверки результатов работы на соответствие установленным требованиям. Непрерывное тестирование ускоряет поставку программного обеспечения, делая весь процесс более быстрым. А благодаря незамедлительной обратной связи, которая помогает уже на самых ранних этапах выявлять ошибки и другие проблемы в приложении, гарантирует, что команды разработки будут создавать высококачественные и надежные приложения. Кроме того, сама способность организовать и проводить эффективное тестирование может значительно снизить затраты в компании, как за счёт экономии времени разработчиков, так и вследствие создания непрерывного конвейера поставки, в котором они могут быстро вносить изменения в код с минимальными рисками нарушения работоспособности приложения в продуктивной среде, где продукт доступен пользователям.

Главным элементом непрерывного тестирования является его автоматизация. К преимуществам автоматизации относятся:

- Аккуратное и тщательное тестирование;
- Высокое покрытие тестами;

- Быстрое обнаружение ошибок;
- Повторное использование тестов;
- Более короткие сроки поставки;
- Экономия времени и денег;

Объектом исследования в данной работе является автоматизированное тестирование программного обеспечения.

Целью работы является разработка структуры написания автоматизированных тестов пользовательского интерфейса программного обеспечения и применение реализованной структуры для регрессионного тестирования системы Интернет-магазин страхового акционерного общества “ВСК”.

Автоматизация тестирования пользовательского интерфейса

Проблематика области автоматизации тестирования

Автоматизированным тестированием называется такой процесс верификации программного обеспечения, в ходе которого основные функции и шаги теста, такие как запуск, инициализация, выполнение, анализ и выдача результата, выполняются автоматически при помощи инструментов для автоматизированного тестирования [1].

Помимо уже названных преимуществ автоматизации тестирования, у данного процесса можно выделить недостатки, к которым относятся:

- необходимость привлечения высококвалифицированного персонала для автоматизации;
- затраты на средства автоматизации, разработку и сопровождение тестов;
- устаревание тестов в случаях изменений требований, переработки интерфейсов тестируемых продуктов и т.п.

Автоматизация тестирования не позволяет полностью исключить ручное тестирование из процесса обеспечения качества, но значительно снижает его долю, тем самым, повышает процент исследовательского тестирования. В рамках обеспечения качества выделяют функциональные, нефункциональные и связанные с изменениями виды тестирования. После проведения необходимых изменений, таких как исправление бага/дефекта, программное обеспечение должно быть перетестировано для подтверждения факта, что проблема была решена. Поэтому, автоматизация наиболее часто применяется при регрессионном тестировании т.е. тестировании, направленном на проверку сделанных изменений.

Учитывая то, что автоматизированный тест является программным продуктом, требующим время на разработку, поддержку и актуализации по

мере необходимости, важно иметь оптимальную архитектуру проекта, удовлетворяющую следующим потребностям:

- Быстрая разработка, которая дает возможность переиспользовать реализованные ранее сценарии;
- Простота поддержки и актуализации. Структура проекта должна обеспечивать быстрый поиск ошибки в коде автоматизируемого сценария и занимать минимальное время на ее исправление;
- Масштабируемость – возможность взаимодействия с различными интеграционными сервисами при введении нового функционала в тестируемую систему.

Пирамида тестирования

На сегодняшний день существуют различные подходы к автоматизации тестирования. Наиболее популярной концепцией является группировки автоматизированных тестов в соответствии “Пирамидой тестирования” [2] (рис. 1). Данная концепция отображает типичное количественное соотношение тестов на каждом из 3-х уровней автоматизации:

1. Unit Tests – модульные тесты, разрабатываемые разработчиками программного обеспечения;
2. Service / API Tests Layer – тесты программного интерфейса приложения;
3. UI Tests – тесты пользовательского интерфейса.

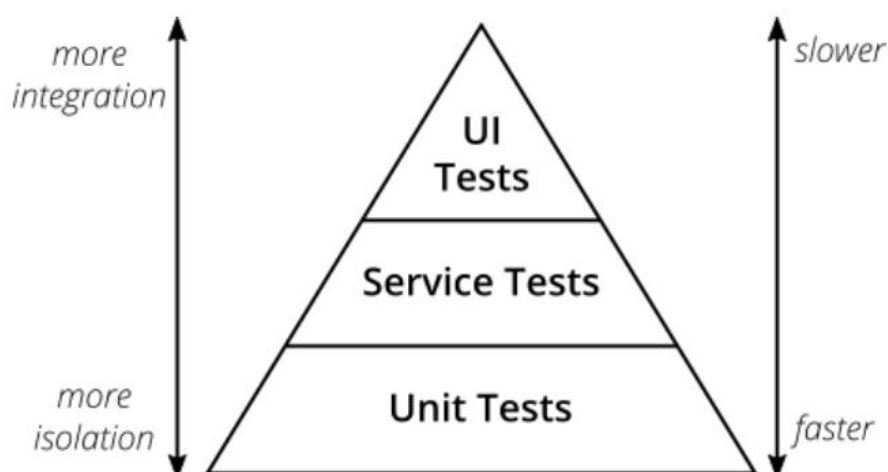


Рисунок 1 - Пирамида тестирования

Актуальным для данной является реализация проекта регрессионного тестирования тестов пользовательского интерфейса. Автоматизация тестирования программных продуктов через графический пользовательский интерфейс набирает всё большую популярность в настоящее время, поскольку только в этом случае приложение будет протестировано тем же самым способом, которым оно будет использоваться конечными пользователями.

Стоит отметить, что данный вид автоматизации тестирования является наиболее сложными и дорогостоящим в разработке. Реализуется такой вид тестирования через использование специализированных инструментов для автоматизации, которые предоставляют возможность взаимодействия с пользовательским интерфейсом тестируемой программы.

Анализ предметной области

Описание тестируемого приложения

Страховое акционерное общество “ВСК” — одна из системообразующих российских страховых компаний. Она является общероссийской универсальной страховой компанией и имеет лицензии на осуществление страховой деятельности по более ста видам страхования физических и юридических лиц, а также оказывает услуги перестрахования. Компания активно развивает внедрение информационных технологий в процессы продаж. Для осуществления обширного набора услуг компания имеет множество информационных систем как внутренних, так и внешних, направленных на конечного пользователя.

В данной работе, в качестве тестируемого приложения рассматривается интернет-магазин “ВСК”. Интернет-магазин “ВСК” — система, реализованная в виде интерактивного веб-сайта, предоставляющая конечному потребителю (физическому лицу) возможность покупки онлайн договоров по многим видам страхования: ОСАГО, Каско, и ДВС (добровольные виды страхования). Система агрегирует в себе следующие компоненты, предоставляемые для взаимодействия с пользователем системы:

- Авторизация пользователя;
- Работа с личным кабинетом пользователя;
- Оформление страховых продуктов;
- Интеграция с платежными шлюзами;

Разработка данной системы является долгосрочным проектом, объемным и сложным по своей структуре. Проект постоянно дорабатывается и меняется: появляется новый функционал, интерфейс приложения адаптируется под потребности пользователей, расширяется линейка предоставляемых продуктов. Поэтому, регрессионное

тестирование данного приложения без внедрения процессов автоматизации может оказаться трудоемким в перспективе многолетней поддержки и развития продукта в условиях конвейерной поставки новых версий в продуктивную среду.

Анализ тестовых сценариев

Тестовый сценарий (Test case) — это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части [3]. Автоматизируемые в данной работе тестовые сценарии реализуют проверки базового процесса оформления, а также процесса пролонгации полисов страхования, данные которых на каждом шаге оформления верифицируются на интеграционном уровне.

В ходе анализа тестовых сценариев были выявлены основные бизнес-процессы работы системы:

- Авторизация в системе, посредством SMS аутентификацией по номеру телефона клиента. На тестовых окружениях авторизационные сообщения хранятся в сервисах информационных шлюзов, предоставляющих программные интерфейсы для взаимодействия;
- Оформление полиса страхования, которое включает заполнение данных: выбор продукта страхования, выбор пакетов страхования, ввод данных страхователя, застрахованных и проведение транзакций по оплате сформированного предложения. В процессе выполнения данных действий, требуется реализация проверок состояния веб-элементов страницы при взаимодействии с ними. Данные о параметрах оформляемого полиса должны соответствовать условиям теста, фиксируемых в тестовом сценарии.

- Проверки наличия купленного полиса в базе данных системы, его отображение в личном кабинете клиента. Интеграционные проверки в системе фискальных платежей, а также других, внешних по отношению к интернет-магазину сервисах.

Таким образом, тестовые сценарии нацелены на воспроизведение действий, имитирующих действия пользователя через пользовательский интерфейс интернет-магазина. Прохождение клиентского пути требует взаимодействие с программными интерфейсами различных компонентов интеграционных систем.

Используемые средства автоматизации тестирования

В контексте автоматизации тестирования веб-приложений, необходим инструмент управления браузером. В качестве такого инструмента наиболее распространенным является – Selenium [4]. Для каждого браузера существует свой драйвер Selenium (например, ChromeDriver для браузера Chrome, GeckoDriver для Mozilla) имеющий уникальную реализацию, однако, для пользователя (тестировщика-автоматизатора) они предоставляют одинаковый набор команд, описанных с учетом выбранного языка программирования (рис. 2).

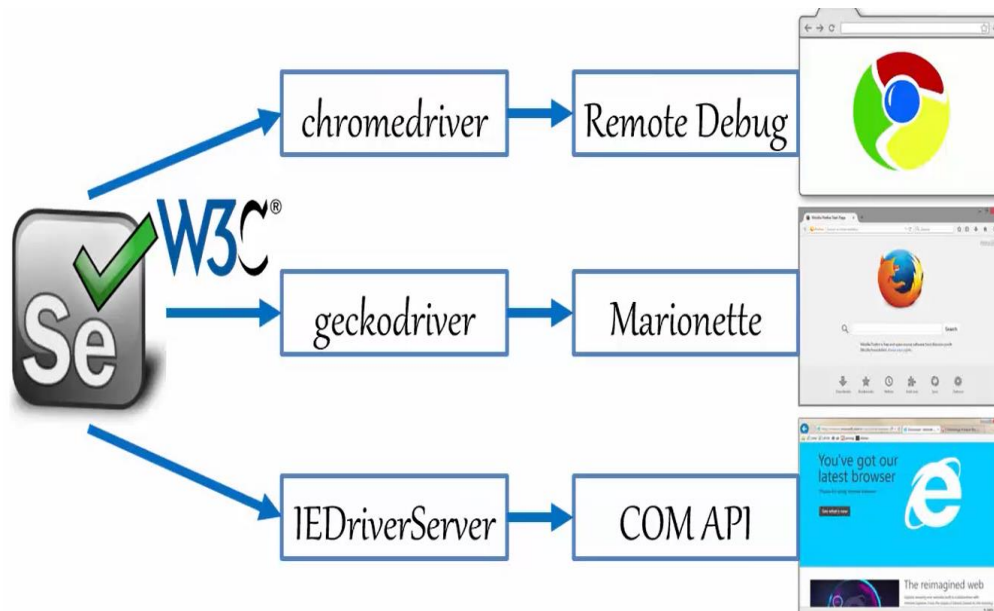


Рисунок 2 - Взаимодействие Selenium с браузерами через соответствующие драйвера

Selenium – продукт с открытым исходным кодом, позволяющий разработчикам браузеров модифицировать текущие решения в зависимости от потребностей конечных потребителей. Поэтому в части предоставляемых драйверов управления браузером данный инструмент является наиболее удобным, а конкурирующие решения предоставляют по большей части надстройки над драйверами Selenium (например, Watir для браузера Internet Explorer).

Основным ограничением данного инструмента является минимальный набор команд по взаимодействию с браузером и веб-элементами на странице. Поэтому, разработчики коммерческих инструментов автоматизации тестирования предлагают ряд конкурирующих решений, которые предоставляют альтернативные к базовому интерфейсу Selenium. Например, такие решения, как Selenide [5], предоставляют расширенный функционал по взаимодействию как с самой оберткой браузера, так и с элементами на страницах. Данные решения накладывают ограничения в используемых технологиях разработки, что является важным фактором неиспользования в долгосрочных

коммерческих проектах. Проанализировав и взвесив все преимущества и недостатки в рамках автоматизации тестирования систем предприятия-заказчика используется базовый интерфейс Selenium по взаимодействию с браузером, а в ходе работы по реализации фреймворка применяются собственные интерфейсы по управлению работой браузера и взаимодействию с элементами форм.

Интеграция результатов выполнения тестов с системой управлением тестирования TestRail осуществляется с помощью инструментов удаленного запуска тестов, реализованного на предприятии. Принцип работы данной системы заключается в сборке кода, выгруженного из системы контроля версий проекта автоматизации и автоматическом запуске реализованных тестовых методов в докер-контейнерах с последующей отправкой результатов теста в систему управления тестированием. В рамках данной работы, инструмент удаленного запуска тестов является сторонним к разрабатываемому проекту автоматизации тестирования. Однако, данное условие накладывает требования к используемому стеку технологий. В качестве платформы разработки используется платформа Microsoft .Net, а в качестве среды тестирования - NUnit. Среда разработки тестов NUnit позволяет организовывать иерархию классов тестовых методов, а также содержит средства проверок, ожидаемых и фактических данных.

Таким образом, для автоматизации используется платформа Microsoft Net Core, среда NUnit, и инструмент управления браузером Selenium. Среда разработки Visual Studio позволяет компактно организовать необходимые инструменты в одном решении с помощью соответствующих NuGet пакетов. В качестве драйвера для запуска тестов используется ChromeDriver, как целевой браузер для веб-приложения, который можно внедрить в проект также с помощью NuGet пакета.

Шаблоны проектирования тестов пользовательского интерфейса

Тестовый сценарий включает в себя, помимо операций инициализации браузера и перехода на нужные страницы, множество действий взаимодействия с UI элементами тестируемой системы. Поэтому написание скрипта тестового сценария только с помощью предоставляемых Selenium команд может вызвать трудности на этапе изменений в тестовом сценарии или его исправлении т.к. такой скрипт состоит из дублирующихся шагов. Поэтому очевидна необходимость применения паттерна проектирования, реализующего принцип разделения действий между методами взаимодействия, с драйвером и спецификацией теста (его логикой). Паттерн Page Object является наиболее популярным подходом для организации архитектуры тестового проекта, направленного на тестирование ПО через какие-либо интерфейсы. Основная идея данного шаблона применительно к веб-тестированию – абстрагирование от отдельных веб-интерфейсов и инкапсуляция функций доступа к элементам интерфейса высшего уровня [6], в то время как на уровень теста выносятся взаимодействие со вспомогательными классами. При этом один и тот же вспомогательный метод может переиспользоваться в тестовых сценариях. Данный подход позволяет в полной мере использовать ООП, посредством которого достигается простота поддержки и актуализации программного кода.

Существует множество вариантов реализации шаблона Page Object. Очень часто встречается разделение на три слоя с соответствующим разделением ответственности (рис. 3) [4]. Слоями являются:

- Тестовые сценарии, описывающие логику теста (спецификация поведения системы);
- Операции уровня бизнес логики;
- Операции уровня интерфейса.



Рисунок 3 - Трехуровневая реализация паттерна Page Object

Таким образом, количество уровней разделения слоев в паттерне Page Object определяется сложностью сценария теста. При усложнении сценариев, из них выделяются вспомогательные функции и формируется уровень бизнес-логики вместе с операциями пользовательского интерфейса. Когда бизнес-логики становится много, начинается декомпозиция этого слоя – выделяются вспомогательные объекты (страницы и локаторы).

Реализация проекта автоматизированных тестов

Применение паттерна Page Object. Уровень страницы. Element Object

В качестве каркаса построения архитектуры автоматизированных тестов взят шаблон трехуровневой модели Page Object. Нижний уровень выбранного шаблона содержит классы, описывающие объекты страниц, а также методы работы с ними. В классах реализованы локаторы конкретных объектов и атомарные действия над ними: поиск, получение свойств, установка значений.

Страницы оформления полисов интернет-магазина содержат множество составных web-элементов, объединенных в единую логическую конструкцию: календари, текстовое поле с выпадающим списком и т.д. Поэтому, при описании элементов страниц, оптимальным решением является применение паттерна Page-Elements, суть которого заключается в создании типизированных оберток над стандартным интерфейсом Selenium IWebElement [7].

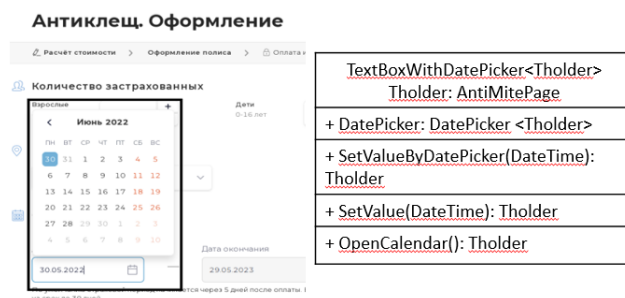


Рисунок 4 - Класс элемента управления текстового поля ввода даты с календарем

В реализованной структуре, классы элементов управления имеют общие принципы взаимодействия с ними. Основным полем в классе элемента управления является корневой элемент типа IWebElement. Для возможности выстраивания цепочек действий над элементами управления

при однократном обращении к экземпляру класса страницы, методы, реализованные в элементах управления, возвращают объект страницы, на которой они расположены. В рамках такой реализации классы являются типизированными (generic), а базовый класс содержит поле Holder принадлежащее к переданному типу страницы:

ControlExBase<THolder>: IWebElement where THolder: PageObjectNotGeneric

Методы по управлению элементом страницы возвращают поле Holder, предоставляя возможность вызова методов из последующих элементов на странице. Таким образом, реализуется возможность последовательного вызова действий с элементами на странице в виде цепочки действий (рис. 5). Такое представление более наглядно демонстрирует структуру страницы и принципы взаимодействия с ней.

```
HypothecCalculationPage
    .OptionStep
    .Region.SelectValueInDropDownAfterTextInput(data.PolicyOfferRegion)
    .CreditDateStart.TextBox.SetValue(data.CreditDateStart.Value)
    .CreditDateEnd.TextBox.SetValue(data.CreditDateEnd.Value)
    .Balance.SetValue((int)data.Balance)
    .Rate.SetValue(data.Rate.ToString().Replace(",", "."))
    .PolicyValidityFromDate.TextBox.SetValue(data.PolicyValidityFrom.Value);
```

Рисунок 5 - Цепочка заполнения данных на 1 шаге оформления полиса

Для инициализации экземпляра класса элемента управления используются фабричный метод [8] (рис. 6). Суть фабричного метода состоит в том, что в зависимости от входного параметра вызывается необходимый конструктор класса, который позволяет создавать типизированные обертки над стандартными элементами управления IWebElement, что обеспечивает возможность использовать элемент управления как относительно DOM модели, так и родительского элемента.

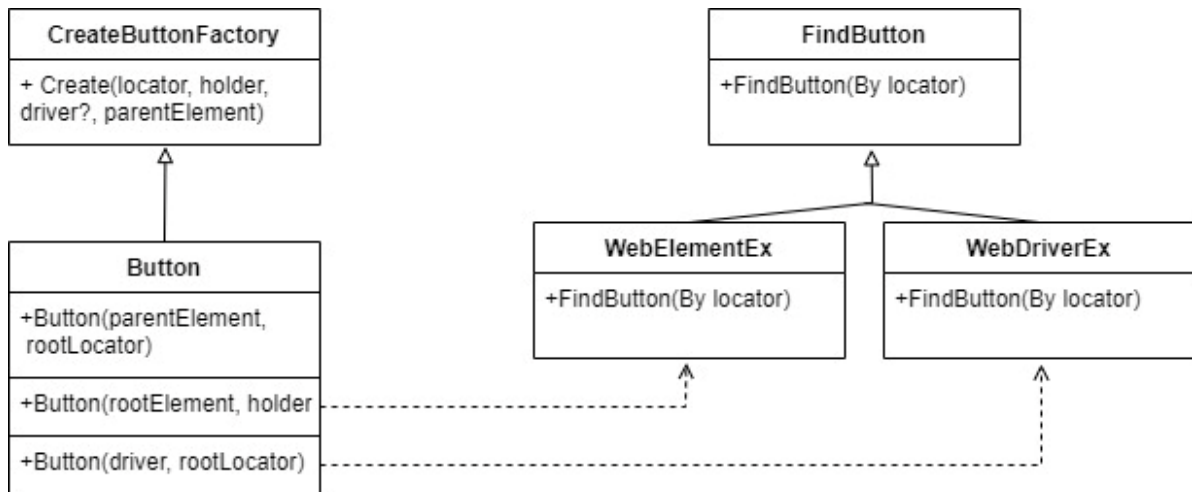


Рисунок 6 - Диаграмма классов фабрики создания элемента управления на примере элемента Button

Анализ тестируемой системы выявил наличие унифицированных групп элементов на страницах оформления продуктов, например, блоки ввода данных страхователей и застрахованного. Для возможности переиспользования данных форм на разных страницах оформления различных продуктов, компоненты агрегированы в соответствующие блоки страниц, где поиск и инициализация элементов управления осуществляется относительно корневого элемента блока.

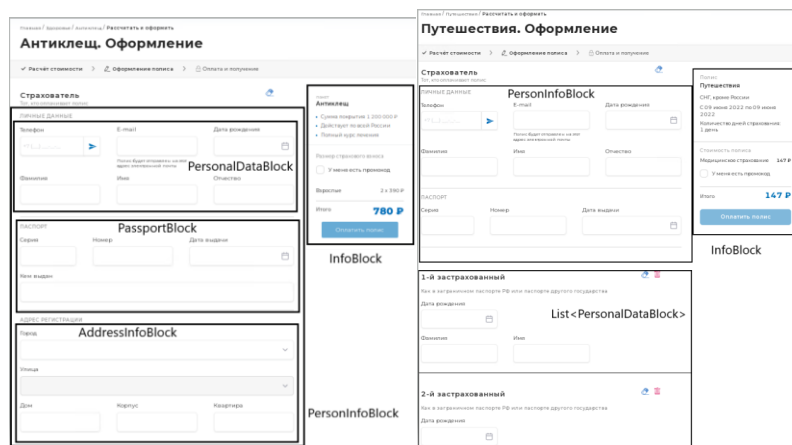


Рисунок 7 - Блочное разделение страниц 2 шага оформления полисов добровольного страхования

Таким образом, применение в совокупности паттернов Page Object и Page Elements, позволяет описывать элементы управления страниц с высокоуровневой абстракцией над базовым интерфейсом Selenium

IWebElement. Данная возможность позволяет реализовывать вспомогательные методы над пользовательскими элементами управления в класса-обертках и приводить описание страниц к компактному виду. Грамотное оперирование локаторами объектной модели html-документа позволяет унифицировать наборы веб-элементов в виде логических блоков, реализуя множественное использование унифицированных наборов на разных страницах.

Операции уровня бизнес-логики

Слой бизнес-логики – уровень, находящийся над уровнем страниц в котором описываются конкретные бизнес-сценарии, интерпретируемые как шаги теста. На данном уровне хранятся классы, описывающие логику ведения тестов с обработкой входных данных, т.е. описываются логически завершенные наборы действий над элементами страниц и производятся валидация фактического и ожидаемого результата. Классы данного уровня содержат в себе экземпляры классов страниц, с которыми осуществляется взаимодействие. Создание экземпляра класса данного уровня реализуется применением паттерна Одиночка. [9] Для этого создан статический класс – MainManager, где заданы инициализации конкретных классов:

```
private HypothecManager _hypothec;  
public HypothecManager HypothecManager  
{  
    get  
    {  
        if (_hypothec == null)  
            _hypothec = new HypothecManager();  
        return _hypothec;  
    }  
    set {hypothec = value;}  
}
```

Благодаря применению данного паттерна, доступ классам уровня бизнес-логики осуществим из любого места в проекте. Помимо этого, класс MainManager предоставляет доступ к браузеру, над которым происходит

управление в тесте. Свойство, хранящее экземпляр активного браузера, обновляется при инициализации веб-драйверов или смене активного браузера управления (метод SwitchTo, рис.9) в базовом классе тестов.

Таким образом, процесс прохождения базового сценария покупки полиса можно разделить последовательностью вызова методов нескольких менеджеров:

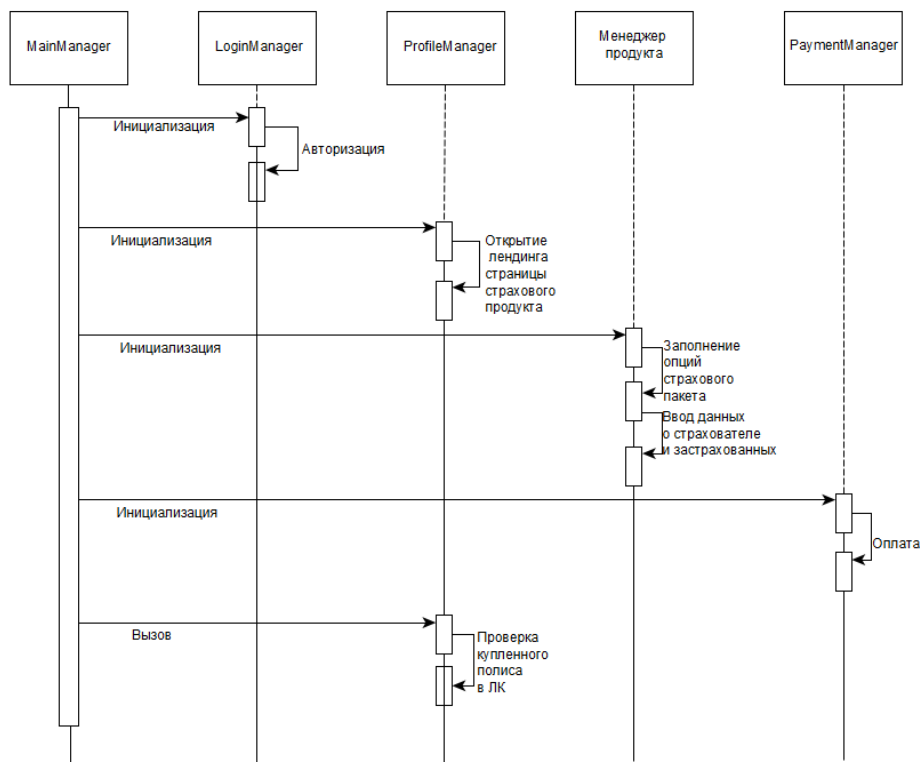


Рисунок 8 - Диаграмма последовательности варианта работы менеджеров при прохождении базового сценария покупки

В итоге реализация унифицированных действий бизнес логики в отдельных программных компонентах тестового проекта, упрощает актуализацию сценария теста при изменении этой логики в системе.

Алгоритм выполнения тестового метода

Принцип применения программной среды тестирования NUnit заключается в выделении классов, реализующих тестовые методы. Платформа NUnit использует систему атрибутов [10], идентифицирующих принадлежность метода или класса его семантике в нотации платформы (табл. 1).

Таблица 1 - Базовые атрибуты платформы NUnit

Атрибут	Описание
TestFixture	Отмечает класс, содержащий тестовые методы
Test	Тестовый метод в классе тестов
SetUp	Метод, исполняемый перед каждым тестовым методом.
TearDown	Метод, вызываемый по завершению исполнения метода теста.
TestCaseSource	Атрибут, используемый в методе параметризованного теста для определения источника данных

Поскольку каждый тестовый метод требует использование драйвера веб-браузера, действия по его инициализации и завершению работы вынесены в методы `SetUp` и `TearDown` соответственно.

Структура данных, используемые в тесте, описываются в соответствующем классе модели данных, которая является входным параметром тестового метода.

Хранение данных для модели теста осуществляется с помощью xml документов, а десериализация этих данных в класс модели осуществляется вспомогательными средствами по работе с xml документами. Так, передавая в атрибут источника данных метод по десериализации файла, платформа NUnit инициализирует переданный список как входные данные тестового метода:

```

public static IEnumerable<AntiMitePolicyData>
PurchaseNoAuthorizedPageDataFromXml() =>
    ModelDataFromXmlFile<AntiMitePolicyData>(dataSubPath +
"PurchaseNoAuthorized.xml");
/// <summary>
/// Тесты 'Покупка. НеАвтр.НетЛК. Web.' [58519], [59532]
/// </summary>
[Test, TestCaseSource(nameof(PurchaseNoAuthorizedPageDataFromXml))]
public static void PurchaseNoAuthorized(AntiMitePolicyData data) { /*Сценарий*/}
    Алгоритм десериализации определен в базовом тестовом классе

/// <summary>
/// Список моделей данных загруженный из XML
/// </summary>
/// <typeparam name="T">Тип модели</typeparam>
/// <param name="fileName">Путь к файлу</param>
protected static IEnumerable<T> ModelDataFromXmlFile<T>(string fileName) where T
: BaseModel
    {
        XmlDocument xml = new XmlDocument();
        xml.Load(TestContext.CurrentContext.TestDirectory + testDataPath +
fileName);
        xml.InnerXml = xml.InnerXml.Replace("BaseTag", typeof(T).Name);

        var loadedList = (List<T>)new
XmlSerializer(typeof(List<T>)).Deserialize(new XmlNodeReader(xml));
        if (loadedList == null)
        {
            throw new Exception($"Не удалось загрузить данные модели из файла
{fileName}");
        }

        return loadedList;
    }

```

Таким образом, подход к хранению данных тестового метода в отдельном файле ускоряет актуализацию теста при изменении данных в тестовом сценарии.

Работа с веб-драйвером браузеров

Для удобного включения механизма инициализации браузера в базовый класс теста, реализована обертка над стандартным интерфейсом Selenium WebDriver. WebDriverWrapper организует хранение информации о параметрах браузера, такой как размер окна, дополнительные опции, режим работы, а также предоставляет методы по его инициализации с

применением заданных параметров. Таким образом, организовывая иерархию базовых тестовых классов, имеется возможность определения конкретных специфичных настроек браузера, каждой для группы тестов.

Ряд сценариев предполагает работу с тестируемой системой одновременно с нескольких устройств. Для выполнения данного требования структура базовых тестовых классов обеспечивает поддержку инициализации и работы с несколькими обертками браузера, а для идентификации конкретного браузера в списке инициализированных определена уникальная роль исполнителя действий (рис. 9).

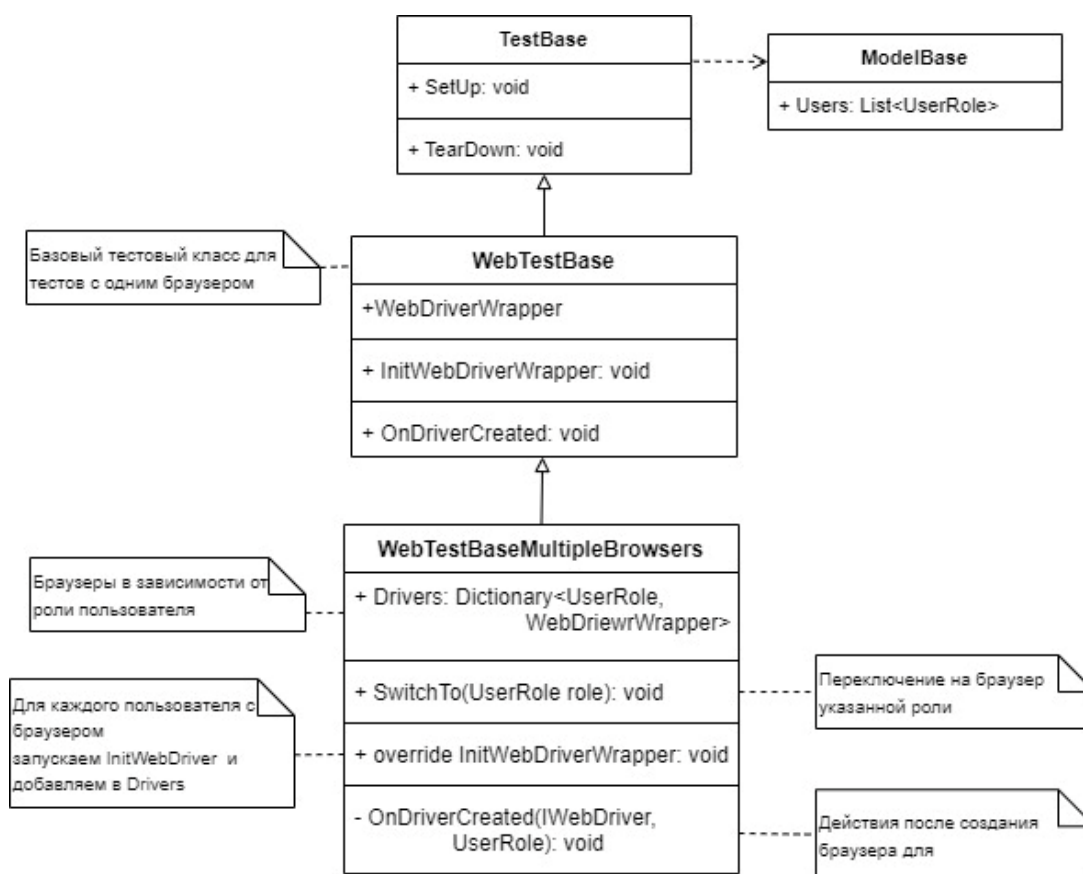


Рисунок 9 — Ролевая схема теста

Данная структура дает возможность точечной настройки браузера для каждой роли, например, включения режима мобильной эмуляции для соответствующего пользователя.

Преимуществом данного подхода является простота расширяемости. При необходимости задания нового пользователя достаточно расширить перечисление ролей и указать список используемых ролей в модели теста.

К недостатку данного подхода можно отнести строгую привязку пользователя к браузеру и отсутствие возможности задания множества браузеров для одной роли – в данном случае придется расширять список ролей до необходимого количества. Однако данный недостаток нельзя назвать критичным т.к. необходимость в неограниченном (параметризованном) количестве браузеров не возникает.

Взаимодействие с API тестируемой системы

Для унификации действий выполнения запросов, логика взаимодействия вынесена в сущности, в которых хранится структура оперируемых в запросах данных. Выполнение REST запросов к системам осуществляется с помощью клиентской Rest API библиотеки платформы Net Core – RestSharp, позволяющей детально настраивать конфигурации отправляемых запросов.

Унификация действий в работе сущностей заключается в выделении абстрактных методов в базовом классе и их переопределение на уровне описываемого запроса:

Таблица 2 – Описание методов и свойств базовой API сущности

Название метода/свойства	Описание
# VerifyCreationOptions: void	Проверка полей, необходимых для выполнения запросов и генерация значений по умолчанию, если они не заполнены
# VariablesValues: Dictionary<string, string>	Значения переменных, передаваемых в запросах с файловым типом данных (XML, JSON)
# MakeRequest: void	Выполнение запросов к сервисам

+ Create: void	Создание API сущности, посредством вызова переопределенных методов.
----------------	---

В ходе анализа тестовых сценариев были выявлены основные бизнес-процессы работы системы, требующие интеграции с API систем:

- Авторизация пользователя в системе, посредством имитации запросов, отправляемых системой ajax запросов;
- Создание полиса страхования во внешних системах, для проверки его интеграции, посредством пролонгации в интернет-магазине;
- Проверка статуса ОФД (оператора фискальных платежей) чека оплаченного полиса;

Реализация авторизации в системе интернет-магазин основана на последовательной отправке запросов, представленных в таблице 3.

Таблица 3 - Описание API методов для авторизации

Название API метода авторизации	Описание
1. postSms	Отправка в сервис авторизации номера телефона, выдача токена взаимодействия с авторизационной сессией
2. checkPostSms	Проверка статуса отправленного SMS
3. postCode	Отправка кода авторизации
4. checkPostCode	Проверка пройденной авторизации (корректности отправленного кода), выдача Cookies авторизации клиенту

Для выполнения авторизации в системе, реализована соответствующая API сущность Authorization [Приложение 3]. В сущности, заданы перечисленные в таблице 3 запросы, которые выполняются последовательно в цепочке вызовов HttpRequestChain метода MakeRequest. Класс Request, как обертка над выполнением запроса предоставляет ряд методов-делегатов по обработке возвращаемых в запросе данных, тем

самым, на уровне формирования запроса указываются действия по их обработке и десериализации ответов в API сущности.

Выполнение авторизации вызовом API методов взаимодействия с системой, вместо выполнения действий по переходу на страницу авторизации и выполнения необходимых действий через веб-интерфейс, позволяет ускорить прохождение тестового сценария, и перейти к основным проверкам сценария сразу после инициализации браузера. Аналогичным образом реализованы сущности создания пролонгируемых полисов и получения данных в сервисе ОФД по купленному полису. Взаимодействие с программными интерфейсами внешних сервисов из автоматизированного теста позволяет внедрять интеграционные проверки в базовые сценарии тестирования, тем самым увеличивая процент покрытия проверок системы в целом.

Модель данных теста

Модель данных — это класс, описывающий структуру данных теста, необходимых для прохождения сценария. Иерархия созданных моделей реализована из расчёта возможности ее расширения и переиспользования в различных сценариях. Базовая модель (рис. 10) служит для агрегации информации, используемой на этапе инициализации браузера, и определения метода заполнения данными по умолчанию, который вызывается непосредственно перед запуском теста в методе Setup (см. табл. 1). Вышестоящие по иерархии модели отвечают за логику выполнения теста. В отдельный класс вынесена модель `PersonalData`, используемая в тестовых сценариях, не требующих прохождения клиентского пути оформления полиса. Модели `PolicyBaseData` и вышестоящие используются в сценариях оформления и содержат специфичные для каждого продукта поля.

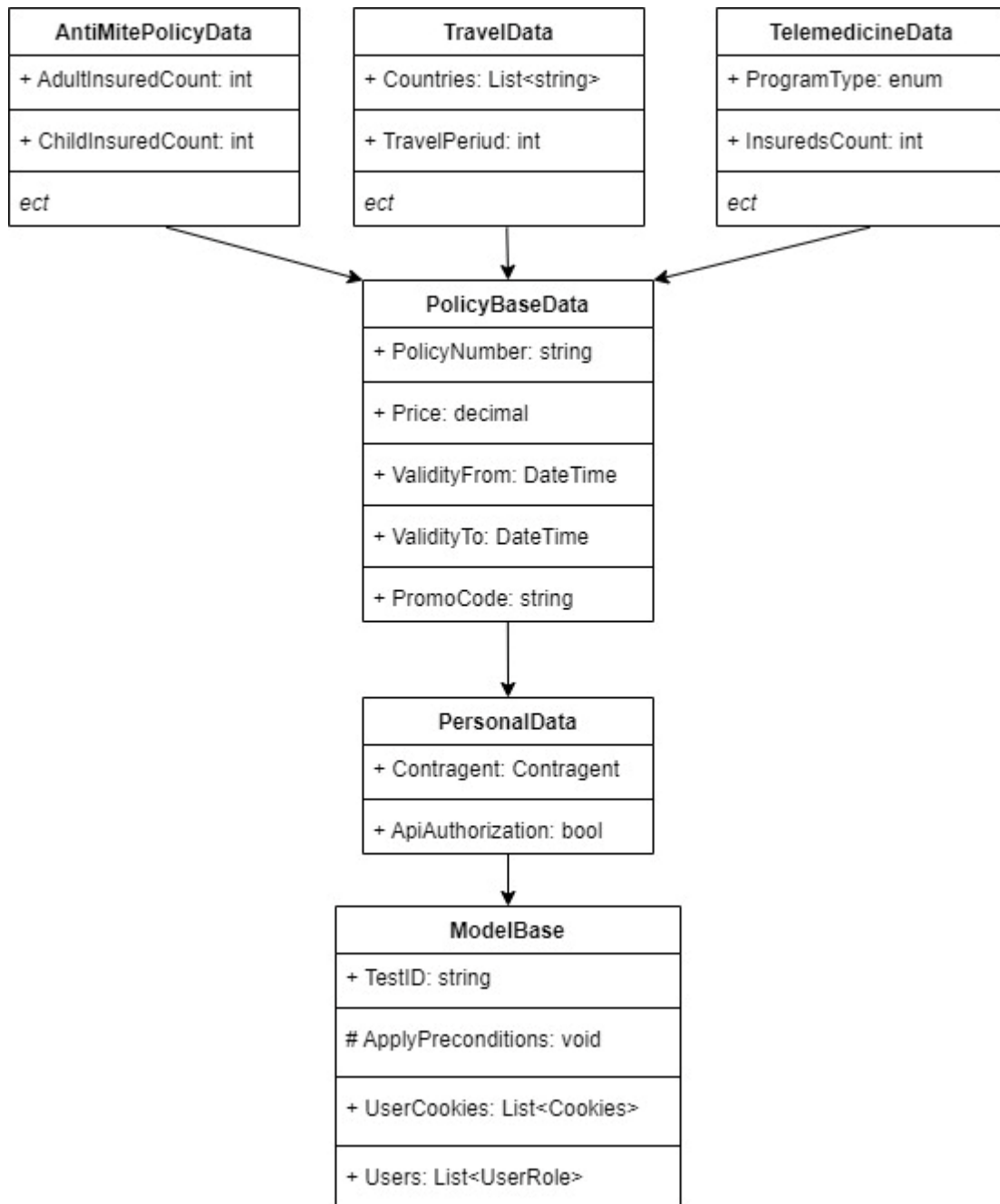


Рисунок 10 — Иерархия моделей данных теста

Таким образом, для исключения дублирования программного кода, при описании структуры данных теста применяется объектно-ориентированный подход с использованием принципов наследования и полиморфизма. Это позволяет повысить эффективность работы со сложной иерархией данных.

Логирование

Логирование — форма автоматической записи в хронологическом порядке операций в ходе выполнения какого-либо процесса. В контексте автоматизированного тестирования важно структурировать и фиксировать выполняемые шаги автоматизированного теста для возможностей:

- Предоставления хронологии выполнения автоматизированного сценария;
- Фиксация ошибки на любом этапе выполнения автоматизированного сценария;
- Получение технической информации о причине “падения” автоматизированного сценария.

Для управления записью информации о происходящих событиях в ходе выполнения сценария, реализован вспомогательный класс работы с логированием [Приложение 4]. Метод для записи событий в файл логирования в качестве аргумента принимает параметр уровня слоя реализованной иерархии, с которого происходит запись. Таким образом, имеется возможность разделения событий по каждому слою, что вносит целостность и унифицированность в каждом лог-файле выполнения теста.

06.06.2022 09:50:55: [T]ШАГ 1: Нажать ссылку 'Личный кабинет'.
06.06.2022 09:50:55: [M]Открытие попапа авторизации
06.06.2022 09:50:55: [F]Кнопка Личный кабинет (логина)
06.06.2022 09:50:55: [R]Клик по кнопке без разрыва цепочки вызовов
06.06.2022 09:50:56: [M]Закреть окно куки
06.06.2022 09:50:56: [R]Клик по кнопке
06.06.2022 09:50:56: [M]Проверка открытия формы логина
06.06.2022 09:50:56: [T]ШАГ 2: Заполнить поле 'Телефон', Нажать 'Получить код'
06.06.2022 09:50:56: [M]Ввод корректного номера (79539276096)
06.06.2022 09:50:56: [R]Устанавливаем значение в контрол 9539276096 ,
06.06.2022 09:50:57: [R]Клик по кнопке без разрыва цепочки вызовов
06.06.2022 09:50:57: [R]Клик по контролю
06.06.2022 09:50:57: [M]Ожидание появления рорир смс
06.06.2022 09:50:59: [M]Проверка элементов попапа авторизации после ввода номера телефона
06.06.2022 09:51:59: [T]ШАГ 3: Ввести в поле полученный 'Смс код'

06.06.2022 09:51:59: [M]Ввод корректного кода номера телефона(79539276096)
06.06.2022 09:51:59: [M]Получаем СМС код для номера 79539276096
06.06.2022 09:51:59: [M]Получаем СМС (объект класса SingleSMS)
06.06.2022 09:52:00: [M]Проверка окна логина после ввода кода
06.06.2022 09:52:00: [T]ШАГ 4: Нажать кнопку "Вход"
06.06.2022 09:52:00: [M]Нажать кнопку "Войти"
06.06.2022 09:52:00: [R]Клик по кнопке без разрыва цепочки вызовов
06.06.2022 09:52:00: [R]Ждем пока откроется личный кабинет
06.06.2022 09:52:29: [M]Проверяем, что открылась страница личного кабинета
06.06.2022 09:52:29: [T]ШАГ 5: В личном кабинете в меню нажать "Вход".
06.06.2022 09:52:29: [M]нажать Войти в ЛК
06.06.2022 09:52:29: [R]Клик по контролю

В целях обеспечения логирования деталей событий с уровня теста и повышения читаемости тестовых методов, реализована специальная нотация написания тестовых методов. Для это реализован статический класс Step, реализующий метод Description, принимающий текстовое описание шага сценария и методы по его выполнению [Приложение 2]. Это помогает организовать удобную структуру на уровне тестового метода, которая обеспечивает удобную навигацию по действиям конкретного сценария. Данная структура упрощает процесс отладки и поиска неисправностей на всех этапах выполнения тестового сценария.

Заключение

В ходе дипломной работы был разработан проект для создания автоматизированных тестов пользовательского интерфейса системы “Интернет-магазин” САО ВСК. Структура данного проекта реализована с использованием шаблона проектирования Page Object. На основании шагов пользовательских сценариев, методы по взаимодействию с бизнес-логикой тестируемой системы были вынесены в отдельные компоненты. Данные компоненты взаимодействуют как с веб-интерфейсом, так и через интеграционный слой, описанный через REST запросы.

Помимо этого, архитектура дополнена функционалом по расширенному логированию, а также имплементирована возможность использования в тестах мультибраузерного взаимодействия.

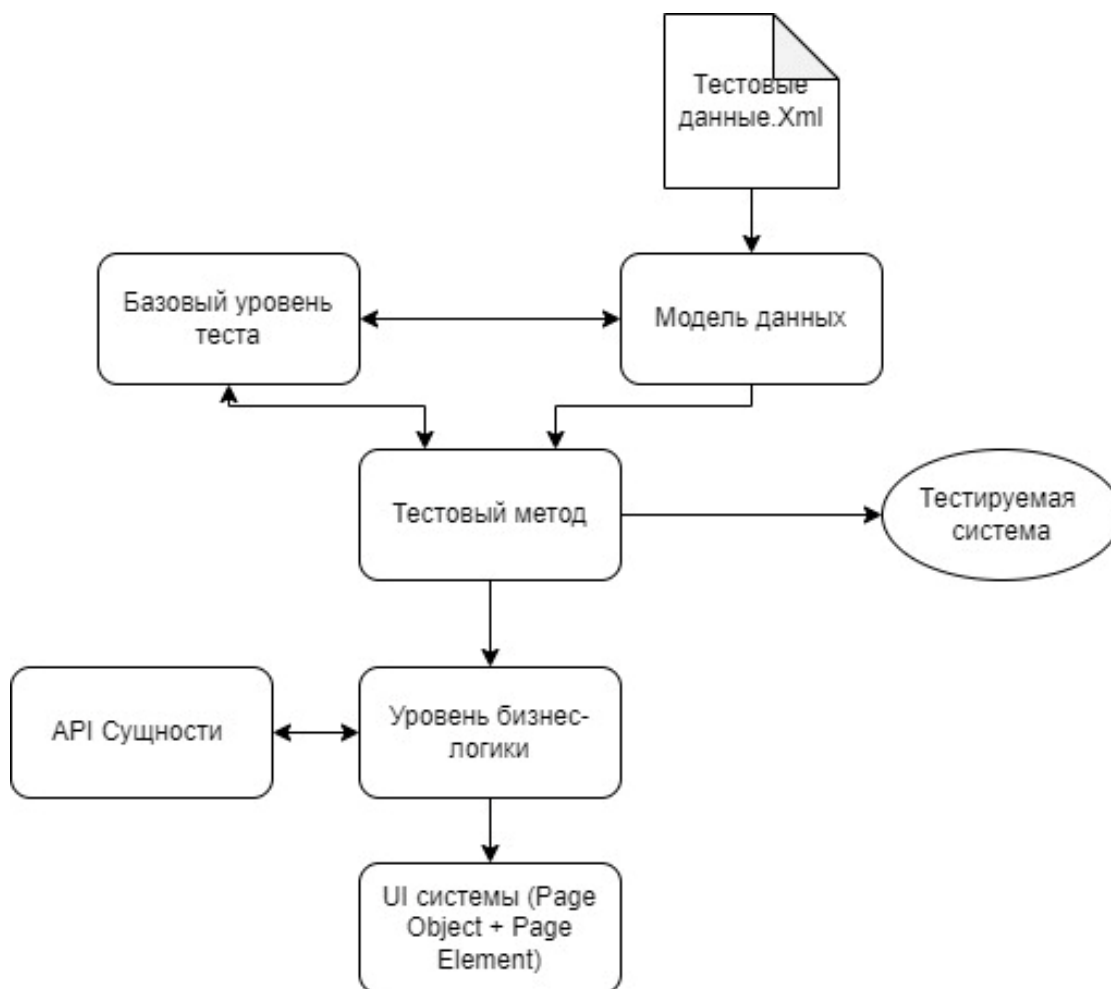


Рисунок 11 - Схема проекта автоматизации тестирования

С помощью разработанной структуры в настоящий момент реализовано порядка 150 тестовых сценариев, нацеленных на проверку процесса оформления полисов ДВС, используемых в регрессионном тестировании системы интернет-магазин. Это подтверждает возможность разработки автоматизированных тестов в данной архитектуре в относительно короткие сроки. Простота актуализации модели данных тестовых сценариев обеспечивается за счет их хранения в структуре отдельных xml файлов. Разделение уровней пользовательского интерфейса системы и бизнес-логики тестовых сценариев позволяет в короткие сроки актуализировать изменения, связанные с доработками в части веб-интерфейса или алгоритма действий определенного шага логики. Взаимодействие с API интеграционных сервисов позволяет расширять тестовые сценарии для тестирования нового функционала системы. Таким

образом, проведенное в данные данной работе исследование, выраженные в разработке программного продукта, позволяют сократить затраты предприятия на ручное тестирование базовых сценариев и повысить долю исследовательского тестирования.

Социальная ответственность

Введение

Научно-исследовательская работа посвящена автоматизации тестирования программного обеспечения программных продуктов компании страхового акционерного общества “ВСК”. Областью применения, разработанного в ходе выполнения работы решения, является процессы разработки ПО, а пользователями данного решения являются тестировщики как часть команды разработки программного продукта. Целью автоматизации тестирования, как и любых других научно-технических решений автоматизации является освобождения человека от участия в процессах получения, преобразования, передачи и использования энергии, материалов, изделий или информации, либо существенного уменьшения степени этого участия или трудоёмкости выполняемых операций. В контексте выполняемой работы цель автоматизации – частичное уменьшение затрат человеческих ресурсов на прохождение однотипных сценариев выявления ошибок в ПО.

Выполнение данной научно-исследовательской работы происходит в офисном здании компании САО “ВСК”, кабинет дирекции тестирования Томского филиала – помещение 3х8 м. Процессы исследования связаны с работой за ПЭВМ. К оборудованию в рабочей зоне относятся ПЭВМ, оснащенное рабочее место для работы за ПЭВМ (стол, кресло), канцелярские принадлежности, системы обеспечения микроклимата.

В данном разделе рассмотрены вопросы, связанные с организацией рабочего места и условий, в которых происходит разработка компонентов автоматизации тестирования.

Правовые и организационные вопросы обеспечения безопасности

Работа на предприятии производится в соответствии с трудовым кодексом РФ. В трудовом кодексе РФ содержатся основные положения отношений между организацией и сотрудниками, включая оплату и нормирование труда, выходных, отпуска и так далее. В соответствии со 137 статьей ТК РФ работник, совмещающий работу с получением высшего образования, имеет право на целевой отпуск за свой счет, предоставляемый нанимателем для:

- прохождения промежуточной аттестации – 15 дней;
- сдачи итоговых экзаменов и создания ВКР – 4 месяца;
- сдачи только итоговых экзаменов – месяц.

Продолжительность рабочего дня работников не должна превышать 40 часов в неделю. Возможно сокращение рабочего времени [12].

Работа в офисе относится ко второй категории тяжести труда – работы выполняются при оптимальных условиях внешней производственной среды и при оптимальной величине физической, умственной и нервно-эмоциональной нагрузки. При работе с персональным компьютером очень важную роль играет соблюдение правильного режима труда и отдыха. В противном случае у персонала отмечаются значительное напряжение зрительного аппарата с появлением жалоб на неудовлетворенность работой, головные боли, раздражительность, нарушение сна, усталость и болезненные ощущения в глазах, в пояснице, в области шеи и руках. В соответствии с ТК РФ Статья 108 в течении рабочего дня работнику должен быть предоставлен перерыв для отдыха и питания продолжительностью не более двух часов и не менее 30 минут. Для указанных условий длительность отдыха при 8-ми часовой рабочей смене составляет 50 минут.

Важной частью работы за ПЭВМ является быстрое и точное считывание информации. Оно обеспечивается при расположении плоскости экрана ниже уровня глаз пользователя,

предпочтительно перпендикулярно к нормальной линии взгляда в 15 градусов вниз от горизонтали). Клавиатура должна располагаться на поверхности стола на расстоянии 100-300 мм от края, обращенного к пользователю. [14]

Схемы размещения рабочих мест с ПК должны учитывать расстояние между рабочими столами с мониторами: расстояние между боковыми поверхностями мониторов не менее 1,2 м, а расстояние между экраном монитора и тыльной частью другого монитора не менее 2 м.

В соответствии с ГОСТ 12.2.032-78. «Система стандартов безопасности труда». Рабочее место при выполнении работ сидя» рабочий стол может быть любой конструкции, отвечающей современным требованиям эргономики и позволяющей удобно разместить на рабочей поверхности оборудование с учетом его количества, размеров и характера выполняемой работы [13]. Выполнение требований на данном рабочем месте отражено ниже в таблице 4.

Таблица 4 - Соблюдение требований рабочего места за ПЭВМ

Требование	Требуемое значение	Значение параметров в помещении
Высота рабочей поверхности, мм, при организации рабочего места.	Нерегулируемая высота (630-755мм)	Нерегулируемая высота (700 мм). Соответствует
Рабочий стул, высота сидения	Высота подколенной ямки над полом, измеренной в положении сидя на плоском сиденье при углах сгибания в коленном и голеностопном суставах 90° с учетом высоты каблука. (430 мм)	Регулируемая высота 410 - 530 мм. Соответствует
Пространство для ног	Расстояние от сиденья до нижнего края рабочей поверхности не менее 150 мм; Ширина не менее 500 мм;	Соответствует

Производственная безопасность

Условия труда, в которых производится работа в офисном помещении предприятия могут спровоцировать появление вредных и опасных факторов производства. В таблице 5 представлены опасные и вредные факторы присутствующие при ведении работы по разработке программного продукта. Данные факторы актуальны на всех этапах работы.

Таблица 5 - Возможные опасные и вредные производственные факторы на рабочем месте оператора ПЭВМ

Факторы (ГОСТ 12.0.003-2015)	Нормативные документы
1. Зрительное напряжение	ГОСТ Р 50948-2001 «Средства отображения информации индивидуального пользования»
2. Отклонение показателей микроклимата в закрытом помещении	СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания"
3. Недостаточная освещенность	СП 52.13330.2016 «Естественное и искусственное освещение». Актуализированная редакция СНиП 23-05-95
4. Нервно-психические перегрузки, монотонность трудового процесса	Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 25.02.2022) (с изм. и доп., вступ. в силу с 01.03.2022)
5. Повышенное значение напряжения в электрической цепи	ГОСТ 12.1.038-82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов.

1) Зрительное напряжение

Длительные зрительные нагрузки (а в рассматриваемом рабочем процессе зрительная нагрузка составляет практически весь рабочий день) могут вызывать ряд патологических состояний зрительного аппарата - Возникновение ложной близорукости, нарушение тонуса глазодвигательных мышц, компьютерный зрительный синдром,

прогрессирование миопии. Для минимизации нарушений зрительного аппарата со стороны видеодисплейного терминала (монитора), дисплей должен соответствовать требованиям, представлен в таблице 6. [21]

Таблица 6 –Регламентируемые параметры плоских дискретных экранов (дисплеев)

Параметр	Допустимые значения
Яркость экрана	не менее 20 кд/м ²
Неравномерность яркости рабочего поля	Не более 20%
Неравномерность яркости элементов экрана	Не более 20%
Яркостный контраст изображения	Не менее 3:1
Ширина контура знака	0,25 – 0,5 мм.
Временная нестабильность изображения (мелькания) для дисплеев	Не должна быть зафиксирована
Частота обновления изображения	Не менее 60 Гц
Пространственная нестабильность изображения (дрожание)	Не более $2 \cdot 10L^{-4L}$, где L - проектное расстояние наблюдения

Профилактические меры по предотвращению расстройств зрения представляют собой элементарные упражнения, помогающие предотвратить развитие спазма аккомодации. Важно периодически переводить взгляд с экрана монитора вдаль, выбрав какой-то предмет в комнате.

2) Отклонение показателей микроклимата в закрытом помещении.

Показатели микроклимата должны обеспечивать сохранение теплового баланса человека с окружающей средой и поддержание оптимального или допустимого теплового состояния организма. Показателями, характеризующими микроклимат в производственных помещениях, являются:

- а) температура воздуха;
- б) температура поверхностей;
- в) относительная влажность воздуха;
- г) скорость движения воздуха;
- д) интенсивность теплового облучения.

Оптимальные микроклиматические условия установлены по критериям оптимального теплового и функционального состояния человека. Они обеспечивают общее и локальное ощущение теплового комфорта в течение 8-ми часовой рабочей смены при минимальном напряжении механизмов терморегуляции, не вызывают отклонений в состоянии здоровья, создают предпосылки для высокого уровня работоспособности и являются предпочтительными на рабочих местах.

Отклонение параметров микроклимата от нормативных значений существенно влияет на здоровье и производительность труда. Высокая температура вызывает интенсивное потоотделение, что приводит к обезвоживанию организма, потере минеральных солей и водорастворимых витаминов С, В1, В2. Нарушение кратности воздухообмена приводит к росту концентрации углекислого газа, что, в свою очередь, негативно сказывается на здоровье сотрудников.

В соответствии с СанПиН 1.2.3685-21 (пункт 5. 29), оптимальные величины параметров микроклимата на рабочих местах применительно к выполнению работ первой категории энергозатрат в холодный и теплый периоды года приведены в таблице 7.

Таблица 7 - Допустимые величины показателей микроклимата на рабочих местах производственных помещений

Период года	Категория работ по уровню энергозатрат, Вт	Температура воздуха, °С		Температура поверхности, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с	
		Диапазон ниже оптимальных величин	Диапазон выше оптимальных величин			для диапазона температур воздуха ниже оптимальных величин, не более	для диапазона температур воздуха выше оптимальных величин, не более
Холодный	Ia (до 139)	20,0 - 21,9	24,1 - 25,0	19,0 - 26,0	15 - 75	0,1	0,1
Теплый	Ia (до 139)	21,0 - 22,9	25,1 - 28,0	20,0 - 29,0	15 - 75	0,1	0,2

В офисных помещениях САО «ВСК» микроклимат поддерживается системой вентиляции и кондиционирования, регулярно проходящей

техническое обслуживание и замер показателей эффективности работы и кратности воздухообмена.

3) Недостаточная освещенность

Выполнение зрительной работы, как с монитором персонального компьютера, так и с бумажными документами при недостаточной освещенности рабочего места может привести к развитию некоторых дефектов глаз: близорукость ложная и истинная; дальнозоркость истинная и старческая, а также к высокой утомляемости и снижению трудоспособности.

В большинстве случаев зрительная работа на ПЭВМ относится к работам высокой точности III-г. Характеристика такой зрительной работы, следующая:

- разряд зрительной работы – III (минимальный размер объекта различения 0,3-0,5мм (толщина символа на экране);

- подразряд работы – «Г», так как фон (экран монитора) — светлый; объект различения (символ, линия, точка и т.п.) — темный, а контраст объекта с фоном — большой.

В соответствии с СП 52.13330.2016 (пункт 4.2), требования, предъявляемые к освещению промышленных помещений, должны удовлетворять показателям, приведенным в таблице 8 (в соответствии с проводимым видом зрительных работ – IIIг).

Таблица 8 - Требования к освещению промышленных помещений предприятий (разряд зрительной работы - IIIг)

Искусственное освещение				Естественное освещение		Совмещенное освещение	
Освещенность, лк		Сочетание нормируемых величин объединенного показателя дискомфорта UGR и коэффициента пульсации		КЕО е _к , %			
при системе комбиниро-	при системе	UGR, не	К _п , не	при верхнем или	При боковом	при верхнем или	при боковом

ванного освещения		общего освещен ия	боле е	боле е	комбинированн ом освещении	освещен ии	комбинированн ом освещении	освещен ии
всег о	В том числе от общег о							
400	200	200	25	15	-	-	3,0	1,2

Работы в здании САО «ВСК» проводятся в помещении с совмещенным освещением. Для регуляции коэффициента естественного освещения применяются раздвижные жалюзи.

4) Нервно-психические перегрузки

Умственное перенапряжение возникает в результате того, что в процессе работы приходится изучать и усваивать большие объемы информации. В силу перегрузки также теряется концентрация сотрудника и способность к эффективному выполнению поставленных задач. В соответствии с ТК РФ, предприятие-работодатель обязано предоставлять работнику во время рабочего дня время на отдых и обед не менее 30 минут [16].

Разработка и реализация алгоритмов также требует большого психоэмоциональных усилий, направленных на концентрацию внимания. Для устранения влияния данного негативного фактора можно применять общепринятые методики – чередование видов умственной нагрузки, перерывы в работе, дыхательная гимнастика.

5) Повышенное значение напряжения в электрической цепи

Офисное помещение предприятия потребляет электроэнергию в больших количествах, обеспечивая бесперебойную и круглосуточную работу компьютеров сотрудников, серверных установок и иных бытовых потребителей. Электричество может являться источником опасности. Несоблюдение правил ГОСТ 12.1.038-82 ССБТ. «Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов» может привести к опасным последствиям. Поражение электрическим током может

произойти при прикосновении к токоведущим частям, находящимся под напряжением, на которых остался заряд или появилось напряжение.

Для переменного тока частотой 50 Гц допустимое значение напряжения прикосновения составляет 2 В, а силы тока – 0,3 мА, для тока частотой 400 Гц, соответственно – 2 В и 0,4 мА, для постоянного тока – 8 В и 1 мА. Мерами защиты от воздействия электрического тока являются оградительные устройства, устройства автоматического контроля и сигнализации, изолирующие устройства и покрытия, устройства защитного заземления, устройства автоматического отключения, предохранительные устройства.

Экологическая безопасность

В офисных помещениях не ведется как такового производства. К отходам, производимым в помещении, можно отнести сточные воды и бытовой мусор. Сточные воды здания относятся к бытовым сточным водам. Основной вид мусора – это отходы печати, бытовой, коробки от техники, использованная бумага. Утилизация отходов печати вместе с бытовым мусором происходит в обычном порядке. Помещение относится к объектам 4 категории оказывающих негативное воздействие на окружающую среду [20, пункт IV].

Предполагаемым источником загрязнения литосферы могут являться образование отходов предметов вычислительной техники при их поломке и прочая оргтехника. Согласно приказу минприроды России от 11.06.2021 N 399 "Об утверждении требований при обращении с группами однородных отходов I - V классов опасности". Юридические лица обязаны сдавать отходы компаниям, которые на основании лицензии ведут деятельность по сбору, транспортировке, обработке, переработке, обезвреживанию и хранению отходов, отнесенных к группе «Оборудование компьютерное, электронное, оптическое, утратившее потребительские свойства».

Накопление и хранение таких отходов должно осуществляться в помещениях, оснащенных воздухообменной вентиляцией и средствами пожаротушения, исключающие доступ к таким отходам посторонних лиц.

Безопасность в чрезвычайных ситуациях

Чрезвычайными ситуациями в рассматриваемых помещениях офисного типа могут быть пожары. Требования по пожарной безопасности устанавливаются ГОСТ 12.1.004-91.

Пожар представляет собой неконтролируемое горение вне специального очага, причиняющее материальный ущерб, вред жизни и здоровью граждан, интересам общества и государства.

С ростом энергооснащенности производства в значительной степени увеличивается опасность пожара в механических мастерских, местах хранения техники и транспортных средств при эксплуатации в них электроустановок. Короткое замыкание, перегрузка, большие переходные сопротивления, взрывы колб и ламп накаливания, замыкания фазных проводов на заземленные конструкции. Чаще всего причиной пожара становится короткое замыкание в электрических установках.

Противопожарная защита должна достигаться применением одного из следующих способов или их комбинацией:

- применением средств пожаротушения и соответствующих видов пожарной техники;
- применением автоматических установок пожарной сигнализации и пожаротушения;
- применением основных строительных конструкций и материалов, в том числе используемых для облицовок конструкций, с нормированными показателями пожарной опасности;

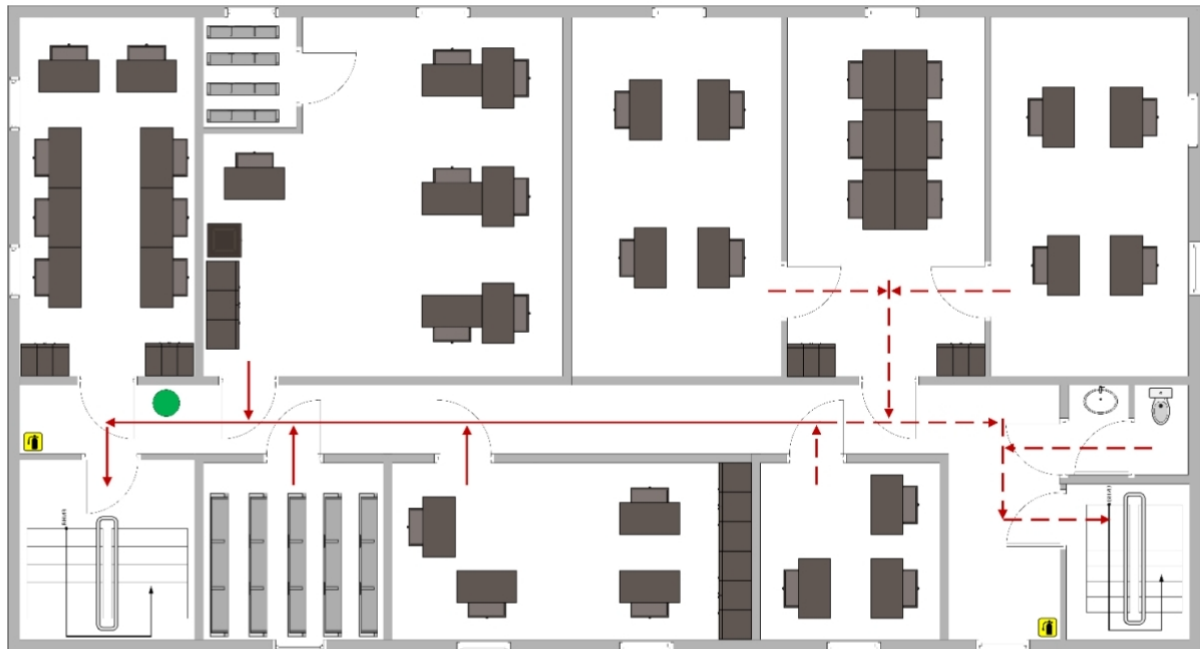
- применением пропитки конструкций объектов антипиренами и нанесением на их поверхности огнезащитных красок (составов);
- устройствами, обеспечивающими ограничение распространения пожара;
- организацией с помощью технических средств, включая автоматические, своевременного оповещения и эвакуации людей;
- применением средств коллективной и индивидуальной защиты людей от опасных факторов пожара;
- применением средств противодымной защиты.
- переходом на пожаробезопасные источники освещения, преимущественно светодиодные.

Для обеспечения эвакуации необходимо:

- установить количество, размеры и соответствующее конструктивное исполнение эвакуационных путей и выходов;
- обеспечить возможность беспрепятственного движения людей по эвакуационным путям;
- организовать при необходимости управление движением людей по эвакуационным путям (световые указатели, звуковое и речевое оповещение и т.п.).

Вещества, материалы и иное производственное оборудование, находящиеся в здании офисного помещения, являются трудно горючими веществами и материалами, поэтому категория помещения по взрывопожарной и пожарной опасности относится к категории пожароопасности В4 [18, пункт 5.1]. В здании находятся 2 эвакуационных выхода, установлена система охраннопожарной сигнализации, имеются в наличии порошковые огнетушители. Доступность эвакуационных выходов регулярно контролируется.

На рисунке 12 представлен план эвакуации при возникновении пожара и других ЧС.



Условные обозначения:

- Вы здесь
- ☒ Огнетушитель
- Путь к основному эвакуационному выходу
- - -> Путь к запасному эвакуационному выходу

Рисунок 12 - План эвакуации помещения CAO "ВСК", ул. Гоголя 35 (2 этаж)

Согласно регламенту пожарной безопасности [22, пункт 8], класс возможного пожара в здании - пожары горючих веществ и материалов электроустановок, находящихся под напряжением – класс пожара Е. В пункте 4.1 свода правил 9.13130.2009 (Техника пожарная. Огнетушитель) указаны меры тушения пожаров электроустановок: порошковыми огнетушителями запрещено тушить электроустановки под напряжением выше 1000 В. А применение воздушно-пенных огнетушителей не разрешено при тушении полностью не обесточенного горящего оборудования. Кроме того, углекислотные огнетушители не применимы при тушении очагов возгорания электрооборудования под напряжением выше 10000 В.

Важный момент при организации тушения пожара в электроустановках – прекращение подачи электрического тока в сеть. Это упрощает процесс пожаротушения, расширяя диапазон применяемых

средств тушения пламени, а также снижает риск дальнейшего распространения пожара и возможности поражения людей, участвующих в борьбе с огнем, электрическим током. Если установка обесточена, ее можно тушить огнетушителями любого типа.

Таким образом к первичным средствам пожаротушения офисного здания можно отнести углекислый и или порошковый огнетушитель.

Вывод

Таким образом, рассмотренные в данном разделе нормативные акты говорят о соответствии нормативным значениям факторов производственной безопасности, защиты окружающей среды и защиты от ЧС. Согласно правилам устройства электроустановок, офисное помещение, в котором проводится работа, относится к 1 категории помещений электробезопасности – помещения без повышенной опасности. Персонал, работающий в офисном помещении, не относится к электротехническому или электротехнологическому персоналу, следовательно, принадлежит к 1 группе по электробезопасности [19]. Согласно СанПиН 1.2.3685-21 работы по разработке программных компонентов относятся к 2 категории тяжести труда ввиду предельно допустимой концентрации внимания и уровнем умственного перенапряжения. Категория помещения по взрывопожарной и пожарной опасности относится к категории В4[18, пункт 5.1]. Также помещение относится к 4 категории объектов, оказывающих негативное воздействие на окружающую среду – минимальное негативное воздействие.

Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

Введение

В данном разделе производится оценка эффективности автоматизации тестирования пользовательских интерфейсов в процессе разработки программного обеспечения, обоснование целесообразности использования для автоматизации тестирования коммерческих средств разработки, выявление сильных и слабых сторон такого решения.

При разработке сложных, многолетних программных продуктов при добавлении даже незначительной части изменений в код программы высок риск возникновения ошибок в старом функционале, поэтому процесс регрессионного тестирования (проверки работоспособности ранее протестированных участков программы) занимает на таких проектах огромное количество времени и человеческих ресурсов.

Кроме того, современный рынок требует от разрабатываемых приложений поддерживать всю линейку современных устройств в огромном количестве конфигураций, тестирование работы приложения на таком количестве конфигураций без использования автоматизированных средств практически нереально.

Внедрение автоматизации тестирования на предприятиях, разрабатывающих программное обеспечение, позволит снизить количество ручного тестирования, улучшить качество приложения и сделать сам процесс разработки быстрым и высокотехнологичным.

Целью данной работы является автоматизация тестирования пользовательского интерфейса на примере системы Интернет-магазин САО “ВСК”.

Для достижения поставленной цели нужно решить следующие задачи:

- Изучение методов и технологий автоматизации тестирования;
- Анализ тестируемого продукта — Интернет-магазина САО “ВСК”;
- Разработка структуры написания теста;
- Разработка автоматизированных тестов в соответствии со структурой;
- Проверка работоспособности реализованных тестов;

1.1 Потенциальные потребители результатов исследования

Для анализа потребителей результатов исследования необходимо рассмотреть целевой рынок и разбить его на сегменты.

Результаты данной разработки будут интересны компаниям производителям программного обеспечения, занимающимся сложными проектами с большим количеством бизнес-функций и значительным сроком жизни приложения (от 1 года и более). К таким типам компаний относятся крупные компании различных секторов:

Таблица 9 – Карта сегментирования рынка

	Программное обеспечение			
	Корпоративный сайт	Информационный портал	Образовательный портал	Интернет-магазины
Крупные компании				
Средние компании				
Стартапы				

1.2 Анализ конкурентных технических решений

В качестве конкурентных решений можно рассмотреть:

1) процесс ручного тестирования, без использования средств автоматизации, при котором тестирование продукта производится командой низкоквалифицированных тестировщиков;

2) процесс тестирования с разработкой автоматических тестов с использованием примитивных инструментов–обёрток над базовым инструментом управления браузером, например Selenide.

В таблице 3 приведено сравнение разработанного в ходе выполнения выпускной квалификационной работы решения с двумя данными альтернативными решениями.

Анализ конкурентных технических решений определяется по формуле:

$$K = \sum_i B_i * B_i$$

где K – конкурентоспособность научной разработки или конкурента;
B_i – вес показателя (в долях единицы); B_i – балл i-го показателя.

Таблица 10 – Оценочная карта сравнения конкурентных решений

Критерий оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б _ф	Б _{к1}	Б _{к2}	К _ф	К _{ф1}	К _{ф2}
Технические критерии оценки ресурсоэффективности							
Функциональные возможности	0.1	5	3	1	0.5	0.3	0.1
Скорость выполнения тестов	0.2	4	2	3	0.8	0.2	0.6
Стабильность результатов	0.2	4	4	3	0.8	0.8	0.6
Простота создания тестов	0.1	3	5	2	0.3	0.5	0.2
Простота поддержки тестов	0.2	4	5	1	0.8	1	0.2
Экономические критерии оценки эффективности							
Затраты на разработку	0.1	3	5	4	0.3	0.5	0.4
Поддержка продукта (процент покрытия)	0.1	5	4	2	0.5	0.4	0.2
	1				4	3.7	2.3

Сильными сторонами разработанного решения являются уровень поддержки продукта и функциональные, нереализуемые в рамках ручного тестирования или использования лишь оберток над базовыми инструментами автоматизации тестирования. Сильной стороной конкурентного решения использовать ручное тестирование является отсутствие необходимости прилагать усилия для поддержания тестов. Использование ручного тестирования выигрывает в сравнении с использованием готовых продуктов для автоматизации, предоставляющих

небогатые возможности и требующих больших усилий для поддержания тестов.

1.4 SWOT анализ

Опишем сильные и слабые стороны проекта, его возможности и угрозы в формате SWOT таблицы (Таблица 11).

Таблица 11 - Матрица SWAT

<p>Сильные стороны проекта С1. Снижение затрат на сотрудников, занимающихся ручным тестированием; С2. Легкость поддержания тестов; С3. Возможность переиспользования тестов, гибкость и масштабируемость; С4. Возможность параллельного запуска тестов на разных браузерах; С5. Сокращение времени на регрессионное тестирование</p>	<p>Слабые стороны проекта Сл1. Высокая стоимость разработки на начальной стадии проекта; Сл2. Зависимость от команд ручного тестирования по части написания сценариев.</p>
<p>Возможности: В1. Развитие IT-бизнеса, рост доходов в компании; В2. Высвобождение средств, которые могут быть использованы на автоматизацию тестирования; В3. Увеличение спроса на разработку крупных программных продуктов;</p>	<p>Угрозы: У1. Появление конкурентных продуктов или технологий для автоматизации. У2. Блокировка иностранных продуктов и технологий для автоматизации на отечественном рынке.</p>

На втором этапе проведения SWOT-анализа проводится составление интерактивных матриц проекта, в которых производится анализ соответствия параметров SWOT каждого с каждым. Соотношения параметров представлены в таблицах 12-15.

Таблица 12 – Интерактивная матрица для сильных сторон и возможностей

Сильные стороны проекта						
Возможности проекта		C1	C2	C3	C4	C5
	B1	+	-	-	-	+
	B2	+	+	+	+	+
	B3	-	+	+	0	+

Таблица 13 – Интерактивная матрица для слабых сторон и возможностей

Слабые стороны проекта			
Возможности проекта		C1	C2
	V1	-	0
	V2	+	+
	V3	0	+

Таблица 14 – Интерактивная матрица для сильных сторон и угроз

Сильные стороны проекта						
Возможности проекта		C1	C2	C3	C4	C5
	У1	-	+	+	-	0
	У2	-	+	-	0	-

Таблица 15 - Интерактивная матрица для слабых сторон и угроз

Слабые стороны проекта			
Возможности проекта		C1	C2
	У1	0	-
	У2	-	-

На основе анализа конкурентных решений была составлена матрица SWOT-анализа, в которой показаны сильные и слабые стороны проекта, возможности и угрозы для разработки. Матрица SWOT представлена в таблице 16.

Таблица 16 – SWOT-анализ

	Сильные стороны проекта	Слабые стороны проекта
	<p>C1. Снижение затрат на сотрудников, занимающихся ручным тестированием;</p> <p>C2. Легкость поддержания тестов;</p>	<p>Сл1. Высокая стоимость разработки на начальной стадии проекта;</p> <p>Сл2. Зависимость от команд ручного тестирования по части написания сценариев.</p>

	<p>С3. Возможность переиспользования тестов, гибкость и масштабируемость;</p> <p>С4. Возможность параллельного запуска тестов на разных браузерах;</p> <p>С5. Сокращение времени на регрессионное тестирование</p>	
<p>Возможности:</p> <p>В1. Развитие IT-бизнеса, рост доходов в компании;</p> <p>В2 Высвобождение средств, которые могут быть использованы на автоматизацию тестирования;</p> <p>В3. Увеличение спроса на разработку крупных программных продуктов;</p>	<p>В3С5 Реализация конвейерной поставки новых версий программного обеспечения в продуктивную среду;</p> <p>В1С3 Расширение фреймворка автоматизации на новые предприятия, где применяется исключительно ручное тестирование.</p>	<p>В2Сл2 Реализация тестовых сценариев с общими шагами для различных автоматизируемых продуктов;</p> <p>В3Сл2 Привлечение ручных тестировщиков к автоматизации сценариев с использованием реализованных программных компонентов.</p>
<p>Угрозы:</p> <p>У1. Появление конкурентных продуктов или технологий для автоматизации.</p> <p>У2. Блокировка иностранных продуктов и технологий для</p>	<p>У1С3 Подготовиться к возможному варианту перехода на другой инструмент, усилить декомпозицию кода, использование абстракций,</p>	<p>У1Сл1 Архивирование программных библиотек, используемых в фреймворке;</p> <p>У2Сл1 Разработка интеграции с отечественными инструментами</p>

автоматизации на отечественном рынке.	чтобы в случае угрозы была возможность переиспользовать бизнес-логику тестов даже при смене инструмента.	управления тестированием.
---------------------------------------	--	---------------------------

2 Определение возможных альтернатив проведения научных исследований

Для определения альтернативных путей проведения научных исследований и вариантов реализации технической части был использован морфологический подход. Созданная морфологическая матрица представлена в таблице 17.

Таблица 17 – Морфологическая матрица

	1	2	3
А. Интерфейс управления браузером	Selenium	Selenide	Protractor
Б. Язык разработки	C#	Python	JavaScript
В. Место запуска тестов	Локально	Удаленный браузер	Виртуальный агент запуска
Г. Разработчик	Высококвалифицированный программист	Программист низкой квалификации	Тестировщик без навыков программирования

Из полученной морфологической матрицы можно выделить 3 варианта реализаций проекта:

- Исполнение 1. А1Б1В1Г1;
- Исполнение 2. А2Б1В2Г2;

- Исполнение 3. АЗБЗВ1Г2;

3.1 Структура работ в рамках научного исследования

Для организации работы в рамках научного исследования необходимо составить полный перечень работ и определить занятость каждого участника в проекте. Такой перечень приведен в таблице 18.

Таблица 18 – Перечень этапов, работ и распределение исполнителей

Основные этапы	№ работ	Содержание работ	Должность исполнителя
Разработка задания	1	Постановка задач	Руководитель
Выбор направления исследования	2	Анализ тестируемого приложения	Инженер
	3	Изучение инструмента автоматизации, технологий	Инженер
Реализация	4	Настройка инструмента	Инженер
	5	Создание проекта	Инженер
	6	Реализация архитектуры тестов	Инженер
	7	Написание тестов внутри созданной архитектуры	Инженер
Тестирование системы	8	Отладка тестов в среде для запуска тестов	Инженер
Анализ и оформление результатов	9	Оценка полученных результатов	Руководитель

3.2 Определение трудоемкости выполнения работ

Трудоемкость выполнения работ рассчитывается по следующим формулам:

$$t_{ожі} = \frac{3t_{\min_i} + 2t_{\max_i}}{5}$$

где $t_{ож}$ – ожидаемая трудоемкость выполнения i -ой работы чел.-дн.;

t_{\min_i} – минимально возможная трудоемкость выполнения заданной i -ой работы (оптимистическая оценка), чел.-дн.;

t_{\max_i} – максимально возможная трудоемкость выполнения заданной i -ой работы (пессимистическая оценка), чел.-дн.

Исходя из ожидаемой трудоемкости работ, определяется продолжительность каждой работы в рабочих днях T_p , учитывающая параллельность выполнения работ несколькими исполнителями. Такое вычисление необходимо для обоснованного расчета заработной платы, так как удельный вес зарплаты в общей сметной стоимости научных исследований составляет около 65 %.

$$T_{pi} = \frac{t_{ожi}}{Ч_i}$$

Где T_{pi} – продолжительность одной работы, раб. дн.;

$t_{ожi}$ – ожидаемая трудоемкость выполнения одной работы, чел.-дн.;

$Ч_i$ – численность исполнителей, выполняющих одновременно одну и ту же работу на данном этапе, чел.

Расчеты продолжительности работ представлены в таблице 19.

3.3 Разработка графика проведения научного исследования

Для сравнительно небольших по объему научных работ, наиболее удобным и наглядным является построение ленточного графика проведения научных работ в форме диаграммы Ганта.

Диаграмма Ганта – горизонтальный ленточный график, на котором работы по теме представляются протяженными во времени отрезками, характеризующимися датами начала и окончания выполнения данных работ.

Для удобства построения графика, длительность каждого из этапов работ из рабочих дней следует перевести в календарные дни. Для этого необходимо воспользоваться следующей формулой:

$$T_{ki} = T_{pi} * k_{\text{кал}}$$

Где T_{ki} – продолжительность выполнения i -й работы в календарных днях;

T_{pi} – продолжительность выполнения i -й работы в рабочих днях;

$k_{\text{кал}}$ – коэффициент календарности.

Коэффициент календарности определяется по следующей формуле:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}}$$

Где $T_{\text{кал}}$ – количество календарных дней в году;

$T_{\text{вых}}$ – количество выходных дней в году;

$T_{\text{пр}}$ – количество праздничных дней в году;

Рассчитанные значения в календарных днях по каждой работе T_{ki} необходимо округлить до целого числа.

Для расчета коэффициента календарности подсчитаем количество рабочих и выходных дней в 2022 году. Всего в году 247 рабочих дней и 118 выходных и праздничных дней. Исходя из полученных данных, рассчитывается коэффициент календарности:

$$k_{\text{кал}} = \frac{T_{\text{кал}}}{T_{\text{кал}} - T_{\text{вых}} - T_{\text{пр}}} = \frac{365}{247} = 1,48$$

На основе таблицы 19 построен календарный план-график проведения работ (рис. 13).

Таблица 19 – Временные показатели проведения научного исследования

	Трудоёмкость работ									Длительность работ в рабочих днях T_{pi}			Длительность работ в календарных днях T_{ki}		
	t_{min} , чел-дни			t_{max} , чел-дни			$T_{ож}$, чел-дни								
	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
Постановка задач	1	1	1	2	2	2	1,4	1,4	1,4	1,4	1,4	1,4	2,07	2,07	2,07
Анализ тестируемого приложения	4	3	3	4	4	3	4	3,4	3	4	3,4	3	5,92	5,03	4,44
Изучение инструмента автоматизации, технологий	5	6	7	10	10	10	7	7,6	8,2	7	7,6	8,2	10,36	11,248	12,136
Настройка инструмента	3	4	5	5	7	7	3,8	5,2	5,8	3,8	5,2	5,8	5,624	7,696	8,584
Создание проекта	5	15	15	10	25	25	7	19	19	7	19	19	10,36	28,12	28,12
Реализация архитектуры тестов	20	5	5	30	10	10	24	7	7	24	7	7	35,52	10,36	10,36
Написание тестов внутри созданной архитектуры	10	25	25	14	45	45	11,6	33	33	11,6	33	33	17,168	48,84	48,84
Отладка тестов в среде для запуска тестов	3	10	10	5	15	15	3,8	12	12	3,8	12	12	5,624	17,76	17,76
Оценка полученных результатов, отчетность	7	7	7	14	14	14	9,8	9,8	9,8	12,32	12,32	12,32	18,2336	18,2336	18,2336
Итого										74,92	100,92	101,72	110,8796	149,3576	150,5436

Исполнители	Ткi, кал. Дн.	Продолжительность выполнения работ											
		Февраль			Март			Апрель			Май		
		1	2	3	1	2	3	1	2	3	1	2	3
Руководитель	2	●											
Инженер	6	■											
Инженер	10		■										
Инженер	6			■									
Инженер	10				■								
Инженер	35,52					■							
Инженер	17,168								■				
Инженер	5,624									■			
Руководитель	12,32										■		

■ Руководитель

■ Студент

Рисунок 13 – Календарный план-график проведения работ (Исп.1, календарные дни)

4.1 Расчёт материальных затрат НТИ

Данная статья включает стоимость всех материалов, используемых при разработке проекта.

Расчет материальных затрат осуществляется по следующей формуле:

$$Z_M = (1 + k_T) * \sum_{i=1}^m C_i * N_{расхи}$$

Где m – количество видов материальных ресурсов, потребляемых при выполнении научного исследования;

$N_{расхи}$ – количество материальных ресурсов i -го вида, планируемых к использованию при выполнении научного исследования (шт, кг, м, м² и т.д.);

C_i – цена приобретения единицы i -го вида потребляемых материальных ресурсов (руб./шт., руб./кг, руб./м, руб./м² и т.д.);

k_T – коэффициент, учитывающий транспортно-заготовительные расходы.

Величина коэффициента (k_T), отражающего соотношение затрат по доставке материальных ресурсов и цен на их приобретение, зависит от условий договоров поставки, видов материальных ресурсов, территориальной удаленности поставщиков и т.д. Транспортные расходы принимаются в пределах 15- 25% от стоимости материалов. Материальные затраты, необходимые для данной разработки, заносятся в таблицу 20.

Среднее потребление электроэнергии офисным компьютером за рабочий день длиной в 8 часов составляет 0,96 кВт*ч.

Таблица 20 – Материальные затраты

Наименование	Ед. измерения	Количество			Цена за ед., руб.	Затраты на материалы, (Зм), руб.		
		Исп.1	Исп.2	Исп.3		Исп.1	Исп.2	Исп.3
Тетрадь для записей	шт.	1			57	57		
Ручка	шт.	1			50	50		
Электроэнергия (с учетом количества рабочих дней)	кВт*ч	71,92	96,88	97,65	3,85	276,9	373	375,95
Итого						383,9	480	482,95

Итого общие материальные затраты составили 356,48 руб.

4.2 Расчёт затрат на специальное оборудование для научных (экспериментальных) работ

В данную статью включают все затраты, связанные с приобретением специального оборудования (приборов, контрольно-измерительной аппаратуры, стендов, устройств и механизмов), необходимого для проведения работ по конкретной теме. Определение стоимости спецоборудования производится по действующим прейскурантам, а в ряде случаев по договорной цене. Расчёт затрат по данной статье приведен в таблице

Таблица 21 – Затраты на специальное оборудование

№ п/п	Наименование оборудования	Кол-во единиц оборудования			Цена единицы оборудования, руб.			Общая стоимость оборудования, руб.		
		Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3	Исп.1	Исп.2	Исп.3
1	Персональный	1			40000			40000		

	компьютер							
2	Мышь	1		700			700	
3	Клавиатура	1		900			900	
4	Интегрированная среда разработки (IDE)	1	0 (бесплатная лицензия)	0 (бесплатная лицензия)	3900	0	0	3900
Итого						40960	40960	44860

В итоге сумма затрат на специальное оборудование составляет 40960 руб.

4.3 Расчёт основной заработной платы исполнителей системы

Данная статья расходов включает основную заработную плату с учетом премий и доплат для исполнителей проекта: студента и научного руководителя. Основная заработная плата ($Z_{\text{осн}}$) руководителя (лаборанта, инженера) рассчитывается по следующей формуле:

$$Z_{\text{осн}} = Z_{\text{дн}} * T_p,$$

где $Z_{\text{осн}}$ – основная заработная плата одного работника;

T_p – продолжительность работ, выполняемых научно-техническим работником, раб. дн.;

$Z_{\text{дн}}$ – среднедневная заработная плата работника, руб.

Среднедневная заработная плата рассчитывается по формуле:

$$Z_{\text{дн}} = \frac{Z_m * M}{F_d},$$

где Z_m – месячный должностной оклад работника, руб.;

M – количество месяцев работы без отпуска в течение года:

при отпуске в 24 раб. дня $M = 11,2$ месяца, 5-дневная неделя;

F_d – действительный годовой фонд рабочего времени научно-технического персонала, раб. дн.

Таблица 22 – Баланс рабочего времени

Показатели рабочего времени	Руководитель	Студент
Календарное число дней	365	365
Количество нерабочих дней:		
- выходные дни	118	118
- праздничные дни		
Потери рабочего времени:		
- отпуск	24	24
- невыходы по болезни		
Действительный годовой фонд рабочего времени	223	223

Месячный должностной оклад работника рассчитывается по следующей формуле:

$$Z_m = Z_{тс} * (1 + k_{пр} + k_d) * k_p,$$

где $Z_{тс}$ – заработная плата по тарифной ставке, руб.;

$k_{пр}$ – премиальный коэффициент, равный 0,3 (т.е. 30% от $Z_{тс}$);

k_d – коэффициент доплат и надбавок;

k_p – районный коэффициент, равный 1,3 (для Томска).

Результаты расчетов основной заработной платы представлены в таблице 23.

Таблица 23 - Расчёт основной заработной платы

Исполнители	$Z_{оклад}$, руб.	$k_{пр}$	k_d	k_p	Z_m , руб	$Z_{дн}$, руб.	T_p , раб. дн.	$Z_{осн}$, руб.
Руководитель	37700	-	-	1,3	49010	2461,49	13,72	33771,63
Инженер	19200	-	-	1,3	24960	1253,6	61,2	76720,10

Таблица 24 - Расчет основной заработной платы с разделением по перечню работ

Название работы	Исполнитель	Трудоемкость, чел.-дн.			Заработная плата, приходящаяся на один чел.-дн., тыс. руб.	Всего заработная плата по тарифу (окладам), тыс. руб.		
		Исп.1	Исп.2	Исп.3		Исп.1	Исп.2	Исп.3
Постановка задач	Руководитель	1,4	1,4	1,4	2461,5	3446,08	3446,1	3446,1
Анализ тестируемого приложения	Инженер	4	3,4	3	1253,6	5014,39	4262,2	3760,8
Изучение инструмента автоматизации, технологий	Инженер	7	7,6	8,2	1253,6	8775,17	9527,3	10279
Настройка инструмента	Инженер	4	5,2	5,8	1253,6	5014,39	6518,7	7270,9
Создание проекта	Инженер	7	19	19	1253,6	8775,17	23818	23818
Реализация архитектуры тестов	Инженер	24	7	7	1253,6	30086,3	8775,2	8775,2
Написание тестов внутри созданной архитектуры	Инженер	12	33	33	1253,6	15043,2	41369	41369
Отладка тестов в среде для запуска тестов	Инженер	4	12	12	1253,6	5014,39	15043	15043
Оценка полученных результатов, отчетность	Руководитель	12,32			2461,5	30325,54		

	Итого	11070 7	14229 8	14330 0
--	-------	------------	------------	------------

4.4 Дополнительная заработная плата исполнителей темы

Данная статья расходов включает заработную плату, начисленную рабочим и служащим не за фактически выполненные работы или проработанное время, а в соответствии с действующим законодательством, в том числе оплата очередных отпусков рабочих, времени, связанного с выполнением государственных и общественных обязанностей. Зная основную заработную плату, можно рассчитать дополнительную заработную плату в размере 13% от основной по следующей формуле:

$$Z_{\text{доп}} = k_{\text{доп}} * Z_{\text{осн}},$$

где $k_{\text{доп}}$ – коэффициент дополнительной заработной платы;

$Z_{\text{осн}}$ – основная заработная плата.

Результаты расчетов дополнительной заработной платы представлены в таблице 25.

Таблица 25 - Дополнительная заработная плата

Исполнитель	Основная заработная плата, руб.			Коэффициент дополнительной заработной платы	Дополнительная заработная плата, руб.		
	Исп. 1	Исп. 2	Исп. 3		Исп.1	Исп.2	Исп.3
Руководитель	31772			0,12	4052	4052	4052
Студент	77723	109314	110316		9327	13118	13238
Итого	10949 5	141086	142088		13379	17170	17290

4.5 Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам государственного социального страхования (ФСС), пенсионного фонда

(ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников. Величина отчислений во внебюджетные фонды определяется исходя из следующей формулы:

$$З_{внеб} = k_{внеб} * (З_{осн} + З_{доп})$$

Где $k_{внеб}$ – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2022 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%.

Отчисления во внебюджетные фонды представлены в таблице 26.

Таблица 26 – Отчисления во внебюджетные фонды

Исполнители	Основная заработная плата, руб		
	Исп.1	Исп.2	Исп.3
Руководитель проекта	35824	35824	35824
Инженер	91102	126484	127606
Итого	126926	162308	163430
Коэффициент отчислений во внебюджетные фонды	0,3		
Исполнение 1	38077,8		
Исполнение 2	48692,4		
Исполнение 3	49029		

4.6 Расчет накладных расходов

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = \frac{\text{сумма статей}}{7} * k_{\text{нр}}$$

где $k_{\text{нр}}$ – коэффициент, учитывающий накладные расходы. Величину коэффициента накладных расходов можно взять в размере 16%. В рассматриваемом случае были использованы 5 статей, поэтому деление производится на 5

$$\text{Для исполнения 1: } Z_{\text{накл}} = \frac{384+40960+109495+13379+38078}{5} * 0,16 = 6473;$$

$$\text{Для исполнения 2: } Z_{\text{накл}} = \frac{480+40960+141086+17170+48692}{5} * 0,16 = 7948;$$

$$\text{Для исполнения 3: } Z_{\text{накл}} = \frac{483+44860+142088+17290+49029}{5} * 0,16 = 8120;$$

4.5 Формирование бюджета затрат научно-исследовательского проекта

На данном этапе сумма затрат по всем статьям расходов рассчитывается и заносится в таблицу 27.

Таблица 27 – Бюджет затрат научно-исследовательского проекта

Наименование статьи	Сумма,руб.			Примечание
	Исп.1	Исп.2	Исп.3	
Материальные затраты НТИ	384	480	483	Пункт 4.1
Расчёт затрат на специальное оборудование для научных (экспериментальных) работ	40960	40960	44860	Пункт 4.2
Расчёт основной заработной платы исполнителей системы	109495	141086	142088	Пункт 4.3
Дополнительная заработная плата исполнителей темы	13379	17170	17290	Пункт 4.4
Отчисления во внебюджетные фонды (страховые отчисления)	38078	48692	49029	Пункт 4.5

Расчет накладных расходов	6473	7948	8120	Пункт 4.6
Бюджет затрат НИИ	208769	256336	261870	Сумма п.1–6

5 Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}}$$

Где $I_{\text{финр}}^{\text{исп.}i}$ – интегральный финансовый показатель разработки;

Φ_{pi} – стоимость i -го варианта исполнения;

Φ_{max} – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

$$\text{Исполнение 1: } I_{\text{финр}} = \frac{208769}{261870} = 0,80$$

$$\text{Исполнение 2: } I_{\text{финр}} = \frac{256336}{261870} = 0,98$$

$$\text{Исполнение 3: } I_{\text{финр}} = \frac{261870}{261870} = 1$$

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное удешевление стоимости разработки в размах.

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i$$

где I_{pi} – интегральный показатель ресурсоэффективности для i -го варианта исполнения разработки;

a_i – весовой коэффициент i -го варианта исполнения разработки;

b_i^a, b_i^p – бальная оценка i -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания (в данном случае 5-ти бальная шкала);

n – число параметров сравнения.

Расчет интегрального показателя ресурсоэффективности проведен в таблице 28

Таблица 28 – Сравнительная оценка

Объект исследования / Критерии	Весовой коэффициент параметра	Исп.1	Исп.2	Исп.3
Функциональные возможности	0,3	5	2	4
Скорость выполнения тестов	0,1	4	4	3
Стабильность результатов	0,2	4	3	5
Простота создания тестов	0,1	3	5	2
Простота поддержки тестов	0,3	5	2	3
Итого	1			

$$I_{p1} = 1,5 + 0,4 + 0,8 + 0,3 + 1,5 = 4,5$$

$$I_{p2} = 0,6 + 0,4 + 0,6 + 0,5 + 0,6 = 2,7$$

$$I_{p3} = 1,2 + 0,3 + 1 + 0,2 + 0,9 = 3,6$$

Интегральный показатель эффективности вариантов исполнения разработки ($I_{исп.i}$) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:

$$I_{исп.1} = \frac{I_{p-исп1}}{I_{финр}^{исп.1}}, \quad I_{исп.2} = \frac{I_{p-исп2}}{I_{финр}^{исп.2}} \text{ и т.д.}$$

$$I_{исп.1} = \frac{4,5}{0,80} = 5,63$$

$$I_{\text{исп.2}} = \frac{2,7}{0,98} = 2,76$$

$$I_{\text{исп.3}} = \frac{3,6}{1} = 3,6$$

Полученное значение интегрального показателя эффективности исполнения разработки (исп.1) выше исполнений конкурентных решений. Таким образом, результат работы можно считать положительным, так как оценка интегрального показателя ресурсоэффективности высокая и это оправдывает использование выбранного решения.

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Сравнительная эффективность проекта ($\mathcal{E}_{\text{ср}}$) определяется формулой:

$$\mathcal{E}_{\text{ср}} = \frac{I_{\text{исп.}i}}{I_{\text{исп.}min}}$$

Таблица 29 – Сравнительная эффективность разработки

№ п/п	Показатели	Исп.1	Исп.2	Исп.3
1	Интегральный финансовый показатель разработки	0,80	0,98	1
2	Интегральный показатель ресурсоэффективности разработки	4,5	2,7	3,6
3	Интегральный показатель эффективности	5,63	2,76	3,6
4	Сравнительная эффективность вариантов исполнения	2,04	1	1,3

Вывод

В ходе оценки перспективности и альтернатив проведения научного исследования с позиции ресурсоэффективности и ресурсосбережения

коммерческого потенциала для выпускной квалификационной работы были определены потенциальные потребители – крупные и среднего размера компании, занимающиеся разработкой веб-систем.

Был сделан анализ конкурентных решений, выявлены сильные и слабые стороны разработанного решения, его возможности и угрозы, а также корреляция этих показателей, определенных в ходе SWOT-анализа.

Исходя из полученных данных и проведенного анализа эффективности, можно сделать вывод, что выбранный вариант исполнения является наиболее эффективным за счет своих возможностей, что делает его разработку целесообразной.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Автоматизированное тестирование программного обеспечения – основные понятия. // ПроТестинг.RU. – [Электронный ресурс] – URL: <http://www.protesting.ru/automation> (дата обращения: 22.05.2022);
2. Пирамида тестирования на практике. //Habr.com. – [Электронный ресурс] – URL <https://habr.com/ru/post/358950/> (дата обращения: 22.05.2022);
3. Тестирование – основные понятия. //ПроТестинг.RU. – [Электронный ресурс] – URL: <http://www.protesting.ru/testing/> (дата обращения: 22.05.2022);
4. Алексей Баранцев. Selenium: полное руководство - 2016 [Электронный ресурс]. – Режим доступа: <https://software-testing.ru/edu/schedule/242> (дата обращения (22.05.2022));
5. Selenide: удобные тесты на Selenium WebDriver. //Habr.com – [Электронный ресурс] – URL <https://habr.com/ru/post/143269/> (дата обращения 23.05.2022);
6. Page Object // martinowler.com – [Электронный ресурс] – URL: <https://martinowler.com/bliki/PageObject.html> (дата обращения 23.05.2022);
7. Уровни абстракции. Создание кастомных элементов. //kreisfahrer.gitbooks.io/selenium-webdriver – [Электронный ресурс] – URL: https://kreisfahrer.gitbooks.io/selenium-webdriver/content/page_object_pattern_arhitektura_testovogo_proekta/urovni_abstraktsii_sozdanie_kastomnih_elementov.html (дата обращения 23.05.2022);
8. Порождающие паттерны. Фабричный метод (Factory Method) //metanit.com сайт о программировании – [Электронный ресурс] –

- URL: <https://metanit.com/sharp/patterns/2.1.php> (дата обращения 23.05.2022)
9. Порождающие паттерны. Одиночка //metanit.com сайт о программировании – [Электронный ресурс] – URL: <https://metanit.com/sharp/patterns/2.3.php> (дата обращения 23.05.2022);
10. NUnit Attribute Description.TestCaseSource //docs.nunit.org – [Электронный ресурс] – URL: <https://docs.nunit.org/articles/nunit/writing-tests/attributes/testcasesource.html> (дата обращения 23.05.2022);
11. Фундаментальная теория тестирования. [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/549054/>
12. Трудовой кодекс Российской Федерации (ред. от 25.02.2022). Статья 173. Гарантии и компенсации работникам, совмещающим работу с получением высшего образования по программам бакалавриата, программам специалитета или программам магистратуры, и работникам, поступающим на обучение по указанным образовательным программам. [Электронный ресурс] – URL http://www.consultant.ru/document/cons_doc_LAW_34683/80ee003dcff42e74722e1f101f707c1561fdfecd/ (дата обращения 12.05.2022);
13. ГОСТ 12.2.032-78 – 2001. Издания. Система стандартов безопасности труда. Рабочее место при выполнении работ сидя. – Введ. 1978. – 04.26 – М.: Из-во стандартов, 2001. – 9 с.
14. ГОСТ 21889-76 – 1993. Издания. Система "Человек-машина". Кресло человека-оператора. Общие эргономические требования. – Введ. 1976 – М.: Из-во стандартов, 1993 – 76 с.
15. СанПиН 2.2.2/2.4.1340-03 (ред. от 25 апреля 2007) “Гигиенические требования к персональным электронно-вычислительным машинам и организации работы”.

- 16.Трудовой кодекс Российской Федерации (ред. от 25.02.2022). Статья 108. Перерывы для отдыха и питания. [Электронный ресурс] – URL http://www.consultant.ru/document/cons_doc_LAW_34683/b25591b628cf4c13d185a767e6c1d04842257a16/ (дата обращения 12.05.2022);
- 17.Свод правил. Естественное и искусственное освещение (изд от. 2017-05-08) [Электронный ресурс] – URL <https://docs.cntd.ru/document/456054197> (дата обращения 12.05.2022)
- 18.Свод правил. 12.13130.2009 Определение категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности. [Электронный ресурс] – URL <https://docs.cntd.ru/document/1200071156> (дата обращения 12.05.2022)
- 19.Правила по охране труда при эксплуатации электроустановок. Приложение к приказу Министерства труда и социальной защиты Российской Федерации от 15 декабря 2020 года N 903н [Электронный ресурс] – URL <https://docs.cntd.ru/document/573264184?marker=6540IN> (дата обращения 12.05.2022)
- 20.Постановление правительства РФ от 31.12.2020 № 2398 “Об утверждении критериев отнесения объектов, оказывающих негативное воздействие на окружающую среду, к объектам I, II, III и IV категорий” [Электронный ресурс] – URL <https://docs.cntd.ru/document/573292854?marker=6540IN> (дата обращения 12.05.2022)
- 21.ГОСТ Р 50948 – 2001. Средства отображения информации индивидуального пользования. Введен 2002-07-01. М.: Стандартинформ, 2008 (переиздание по состоянию на апрель 2008 г.)

22. Технический регламент о требованиях пожарной безопасности. (с изменениями на 30 апреля 2021 года) [Электронный ресурс] – URL <https://docs.cntd.ru/document/902111644>

Приложение 1

```
<?xml version="1.0" encoding="utf-8" ?>
<ArrayOfBaseTag xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <BaseTag>
    <CreditDateStart>
      <Unit>Day</Unit>
      <Count>10</Count>
    </CreditDateStart>
    <CreditDateEnd>
      <Shifts>
        <SingleShift>
          <Unit>Day</Unit>
          <Count>11</Count>
        </SingleShift>
      </Shifts>
    </CreditDateEnd>

    <PolicyValidityFrom>
      <Unit>Day</Unit>
      <Count>59</Count>
    </PolicyValidityFrom>

    <Balance>1000</Balance>
    <Rate>0.5</Rate>

    <RiskList>
      <HypotheRiskType>Страхование жизни</HypotheRiskType>
    </RiskList>
    <CreditDocumentNumber>a33-02/П</CreditDocumentNumber>
    <ApiAuthorization>true</ApiAuthorization>
    <Contragent>
      <CorporateProgram>>false</CorporateProgram>
      <EmailConfirm>true</EmailConfirm>
      <PersonInfo>
        <Gender>>false</Gender>
        <BirthDate>
          <Unit>Year</Unit>
          <Count>-27</Count>
        </BirthDate>
        <Address>
          <Region>Москва</Region>
          <City>Москва</City>
          <Street>1812 года</Street>
          <House>1</House>
          <Building></Building>
          <Apartment>1</Apartment>
        </Address>
        <DocumentInfo>
          <DocumentType>Паспорт гражданина РФ</DocumentType>
          <IssuedBy>УФМС России по Томской области в г.
Томске</IssuedBy>
        </DocumentInfo>
      </PersonInfo>
    </Contragent>
  </BaseTag>
</ArrayOfBaseTag>
```

Приложение 2

```
Step.Description(@"Заполнить все поля данными из Preconditions.",
() =>
{//4
    Main.HypotheсManager
        .ModifyCreditDateEndByFirstAvailableValue(data)
        .FillCalculationBlock(data)
        .Validate.NextStepButtonActive();
});
Step.Description(@"Нажать кнопку ""Рассчитать"". ",
() =>
{//5
    Main.HypotheсManager
        .CalculateAndRememberPaymentData(data)
        .Validate.PriceDisplayed()
        .Validate.NextStepButtonActive();
});
Step.Description(@"Нажать кнопку ""Оформить полис"". ",
() =>
{//6
    Main.HypotheсManager
        .HypotheсCalculationPage
        .NextStepButton.Click();
    Main.CalculationManager
        .ValidateStepOpend(OfferSteps.Insurant);
    Main.HypotheсManager
        .Validate.InsurerLKData(data);
    Main.CalculationManager
        .Validate.InsurerLKPassportData(data.Insurer.Passport);
});
Step.Description(@"Выставить один чек бокс:
- Согласен с Декларацией Застрахованного (жизнь)
Заполнить ""Номер кредитного договора"" более 5 символов.
() =>
{//9
    Main.HypotheсManager
        .FillRestOfForm(data)
        .Validate.NextStepButtonActive();
});
```

Приложение 3

```
public class Authorization : APICallerEntity<Authorization>
{
    /// <summary>
    /// Токен
    /// </summary>
    public string Token;
    /// <summary>
    /// Статус
    /// </summary>
    public string Status;
    /// <summary>
    /// Ответ
    /// </summary>
    public string Response;
    /// <summary>
    /// Номер телефона для авторизации
    /// </summary>
    public string PhoneNumber { get; set; }
    /// <summary>
    /// Куки для авторизации
    /// </summary>
    public List<System.Net.Cookie> AuthorizationCookie;

    protected override void VerifyCreationOptions(APIClientManager apiClientManager)
    {
        if (string.IsNullOrEmpty(PhoneNumber))
        {
            throw new Exception("Не задан номер телефона");
        }
    }

    protected override Authorization MakeRequests()
    {
        Step.Description("Авторизация: " + PhoneNumber,
            () =>
            {
                apiClientManager.GetApiClient("IM").ClearCookie();
                if (AuthorizationCookie == null)
                {
                    var Chain = new HttpRequestChain();
                    Chain
                        .Add((R) => GetSessionID())
                        .Add((R) => PostSmsForLogin(PhoneNumber))
                        .Add((R) => CheckPostSmsForLogin())
                        .Add((R) => PostCodeForLogin(
                            Utils.SMSHelper.GetSMSCode(PhoneNumber)))
                        .Add((R) => CheckPostCode())
                        .Add((_) => ConfirmLogin())
                        .Execute(apiClientManager);

                    Retry.NTimes(() => AuthorizationCookie =
                        apiClientManager.GetApiClient("IM").CookiesList,
                        checkMethod: () =>
                        {
                            return AuthorizationCookie.Count > 0;
                        }, interval: TimeSpan.FromSeconds(2), maxRetryCount: 5);
                }
            }, 0);
        return this;
    }

    /// <summary>
    /// Получение ID Сессии
    /// </summary>
    protected Request GetSessionID()
    {
        return new Request()
        {
            EntityName = "IDSession",

```

```

        ApiType = ApiType.IM,
        Endpoint = "personal/",
        RequestType = RequestType.GET,
        HttpRequestType = HttpRequestType.GET
    };
}
/// <summary>
/// Отправка СМС для авторизации Пользователя
/// </summary>
/// <param name="phone">Номер телефона для авторизации</param>
protected Request PostSmsForLogin(string phone)
{
    return new Request()
    {
        EntityName = "Отправка СМС для Авторизации",
        ApiType = ApiType.IM,
        Endpoint = "ajax/auth/postSmsX/",
        FormParameters = new Dictionary<string, string> { { "phone", $"{phone}" } },
        RequestType = RequestType.FORM,
        ErrorInResponse = response => { return response == ""; },

        OnSuccessJSON = json =>
        {
            Token = (string)json.SelectToken("token");
            Status = (string)json.SelectToken("status");
            Response = json.ToString();
            if (Response.Contains("ERR"))
            {
                var time = Response.Substring(" [\\d]+ ");
                Thread.Sleep(TimeSpan.FromSeconds(time.ToDouble() + 1));
                PostSmsForLogin(PhoneNumber).Execute(apiClientManager);
            }
        }
    };
}
/// <summary>
/// Проверка отправки СМС для авторизации
/// </summary>
protected Request CheckPostSmsForLogin()
{
    return new Request()
    {
        EntityName = "CheckPostSmsForLogin",
        ApiType = ApiType.IM,
        Endpoint = "ajax/auth/checkPostSms/",
        FormParameters = new Dictionary<string, string> { { "token", Token } },
        RequestType = RequestType.FORM,
        ErrorInResponse = response => { return response.Contains("[") || response ==
"; },

        RetryOnError = true
    };
}
/// <summary>
/// Отправка кода для Авторизации
/// </summary>
protected Request PostCodeForLogin(string sms)
{
    return new Request()
    {
        EntityName = "PostCodeForLogin",
        ApiType = ApiType.IM,
        Endpoint = "ajax/auth/postCode/",
        FormParameters = new Dictionary<string, string> { { "pass", sms } },
        RequestType = RequestType.FORM,
        OnSuccessJSON = json =>
        {
            Token = (string)json.SelectToken("token");
            Status = (string)json.SelectToken("status");
            Response = json.ToString();
        }
    };
}

```

```

}
/// <summary>
/// Проверка успешной отправки кода
/// </summary>
/// <returns></returns>
protected Request CheckPostCode()
{
    return new Request()
    {
        EntityName = "ClientCheckPostCode",
        ApiType = ApiType.IM,
        Endpoint = "ajax/auth/checkPostCode/",
        FormParameters = new Dictionary<string, string> { { "token", Token } },
        RequestType = RequestType.FORM,
        ErrorInResponse = response => { return response.Contains("[]") || response ==
"; },
    };
}
}

```



```

/// <summary>
/// Методы логирования для конкретного потока
/// </summary>
public class ThreadLogger
{
    /// <summary>
    /// ИД кейса
    /// </summary>
    private string CaseId;

    /// <summary>
    /// Каталог лога автотеста
    /// </summary>
    public string TestLogDirectory { get; private set; }

    /// <summary>
    /// Путь к файлу лога
    /// </summary>
    public string LogFilePath { get; private set; }

    public ThreadLogger(string caseId)
    {
        CaseId = caseId;
        TestLogDirectory = Path.Combine(LogDirectory.TestAppLogDirectory,
Guid.NewGuid().ToString());
        string fileName = $"[AT]{CaseId}_{DateTime.Now:yyyy_MM_dd}.log";
        Directory.CreateDirectory(TestLogDirectory);
        LogFilePath = Path.Combine(TestLogDirectory, fileName);
    }

    /// <summary>
    /// Запись в лог
    /// </summary>
    /// <param name="message">Сообщение в лог</param>
    /// <param name="level">уровень сообщения</param>
    public void Trace(string message, Level level)
    {
        if (message == null || message.Trim() == "")
        {
            return;
        }

        switch (level)
        {
            case Level.Page: message = " [P]" + message; break;
            case Level.Manager: message = " [M]" + message; break;
            case Level.Test: message = " [T]" + message; break;
            case Level.Start: message = "[S]" + message; break;
            case Level.Finish: message = " [Fin]" + message; break;
        }

        WriteToFile(message);
    }

    /// <summary>
    /// Разделитель
    /// </summary>
    public void Separator()

```

```
{
    WriteToFile(
        "-----");
}
/// <summary>
/// Запись строки в файл
/// </summary>
/// <param name="message"></param>
private void WriteToFile(string message)
{
    File.AppendAllText(LogFilePath, $"{DateTime.Now:dd.MM.yyyy HH:mm:ss}:
{message} \r\n");
}
}
```