

**Министерство науки и высшего образования Российской Федерации**  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Национальный исследовательский Томский политехнический университет»

---

Инженерная школа информационных технологий и робототехники  
Подразделение ОИТ  
Направление, специальность 09.04.02 «Информационные системы и технологии»

### МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка модуля интеграции корпоративной базы знаний на платформе XWiki и системы трекинга задач YouTrack.

УДК 004.4:004.75:519.72

#### Студент

Группа	ФИО	Подпись	Дата
8ИИМ71	Дергачев Антон Олегович		

#### Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мирошниченко Евгений Александрович	К.Т.Н.		

#### Консультант

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель	Копнов Максим Валерьевич	К.Т.Н.		

#### КОНСУЛЬТАНТЫ ПО РАЗДЕЛАМ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОСГН	Сосковец Любовь Ивановна	Д.И.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД	Атепаева Наталья Александровна			

#### ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Савельев Алексей Олегович	К.Т.Н.		

**ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП ПРОГРАММЫ  
МАГИСТРОВ 09.04.02 «ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ»**

Код	Результат обучения (выпускник должен быть готов)
<b>Общепрофессиональные и профессиональные компетенции</b>	
P1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
P2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
P3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
P4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
P5	Разрабатывать стратегии и цели проектирования, критерии

Код	Результат обучения (выпускник должен быть готов)
	эффективности и ограничения применимости, новые методы, средства и технологии проектирования геоинформационных систем (ГИС) или промышленного программного обеспечения.
Р6	Планировать и проводить теоретические и экспериментальные исследования в области создания интеллектуальных ГИС и ГИС технологии или промышленного программного обеспечения с использованием методов системной инженерии.
Р7	Осуществлять авторское сопровождение процессов проектирования, внедрения и сопровождения ГИС и ГИС технологий или промышленного программного обеспечения с использованием методов и средств системной инженерии, осуществлять подготовку и обучение персонала.
Р8	Формировать новые конкурентоспособные идеи в области теории и практики ГИС и ГИС технологий или системной инженерии программного обеспечения. Разрабатывать методы решения нестандартных задач и новые методы решения традиционных задач. Организовывать взаимодействие коллективов, принимать управленческие решения, находить компромисс между различными требованиями как при долгосрочном, так и при краткосрочным планировании.
Общекультурные компетенции	
Р9	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
Р10	Свободно пользоваться русским и иностранным языками как средством делового общения.

Код	Результат обучения (выпускник должен быть готов)
P11	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
P12	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Школа Инженерная школа информационных технологий и робототехники  
Направление подготовки 09.04.02 «Информационные системы и технологии»  
Отделение школы (НОЦ) Отделение информационных технологий

УТВЕРЖДАЮ:  
Руководитель ООП  
\_\_\_\_\_ Савельев А.О.  
(Подпись) (Дата) (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Магистерской диссертации

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

Студенту:

Группа	ФИО
8ИМ71	Дергачеву Антону Олеговичу

Тема работы:

Разработка модуля интеграции корпоративной базы знаний на платформе XWiki и  
системы трекинга задач YouTrack.

Утверждена приказом директора (дата, номер)	05.03.2019 г., № 1734/с
---	-------------------------

Срок сдачи студентом выполненной работы:	03.06.2019
--	------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<b>Исходные данные к работе</b> <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>	Объектом исследования является вики- система XWiki, система отслеживания задач YouTrack, и способы их интеграция.
--	---

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><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>Выявление цели и задач;</p> <p>Изучение предметной области;</p> <p>Разработка спецификации требований;</p> <p>Проектирование решения интеграции вики-системы XWiki и системы отслеживания задач YouTrack при помощи модуля расширения XWiki и веб-сервиса;</p> <p>Выбор технологий реализации;</p> <p>Реализация спроектированного решения;</p> <p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение;</p> <p>Социальная ответственность.</p>
<p><b>Перечень графического материала</b></p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<ul style="list-style-type: none"> <li>• Схема интеграции XWiki и YouTrack;</li> <li>• Детальная схема жизненного цикла модуля расширения;</li> <li>• IDEF-0 диаграмма процесса интеграции</li> <li>• Инфологическая модель (Уровень определений);</li> <li>• Инфологическая модель (Уровень ключей);</li> <li>• Инфологическая модель (Уровень атрибутов);</li> <li>• Функциональная диаграмма IDEF-0</li> <li>• Модель данных вики системы XWiki;</li> <li>• Алгоритм передачи данных: создания, обновления, удаления требований в проекте;</li> <li>• Панель менеджера расширений в XWiki;</li> <li>• Презентация в формате *.pptx.</li> </ul>
<p><b>Консультанты по разделам выпускной квалификационной работы</b></p> <p><i>(с указанием разделов)</i></p>	
<p><b>Раздел</b></p>	<p><b>Консультант</b></p>
<p>Экономическая часть</p>	<p>Ст. преподаватель ОСГН Потехина Нина Васильевна</p>
<p>Социальная ответственность</p>	<p>Ст. преподаватель ООД Атепаева Наталья Александровна</p>
<p>Английская часть</p>	<p>Доцент ОИЯ Сидоренко Татьяна Валерьевна</p>

**Названия разделов, которые должны быть написаны на русском и иностранном языках:**

3. Разработка программного продукта (Development of software product)

**Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику**

28.01.2019

**Задание выдал руководитель / консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мирошниченко Е.А.	к.т.н.		
Ст. преподаватель	Копнов М.В.	к.т.н.		

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ИМ71	Дергачев А.О.		

**Министерство науки и высшего образования Российской Федерации**  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Национальный исследовательский Томский политехнический университет»

---

Инженерная школа информационных технологий и робототехники  
Подразделение ОИТ  
Уровень образования Магистратура  
Направление, специальность 09.04.02 «Информационные системы и технологии»  
Период выполнения Осенний/весенний семестр 2018/2019 учебного года

Формат представления работы:

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

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН**  
выполнения выпускной квалификационной работы

Срок сдачи студентом выполнения работы:	03.06.2019
---	------------

Дата контроля	Название раздела (модуля) виды работы (исследования)	Максимальный балл раздела (модуля)
10.03.2019	Теоретический анализ предметной области и постановка задач	20
15.04.2019	Проектирование модуля расширения	30
21.05.2019	Реализация модуля расширения	25
29.05.2019	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	15
16.05.2019	Социальная ответственность	5
29.05.2019	Обязательное приложение на иностранном языке	5

**СОСТАВИЛ:**  
**Руководитель ВКР**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мирошниченко Е.А.	к.т.н.		

**Консультант**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ст. преподаватель	Копнов М.В.	к.т.н.		

**СОГЛАСОВАНО:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Савельев А.О.	к.т.н.		



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

Студенту:

<b>Группа</b>	<b>ФИО</b>
8ИМ71	Дергачев Антону Олеговичу

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение (НОЦ)</b>	<b>Отделение информационных технологий</b>
<b>Уровень образования</b>	Магистратура	Направление/специальность	09.04.02 «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>	Оклад руководителя - 29568 руб. Оклад студента - 21760 руб. Стоимость материальных ресурсов определяется на основе открытых прайсов.
2. <i>Нормы и нормативы расходования ресурсов</i>	–
3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>	Коэффициент отчислений на уплату во внебюджетные фонды 30 %

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. <i>Предпроектный анализ</i>	Определение потенциальных потребителей, анализ конкурентных решений, SWOT-анализ.
2. <i>Разработка устава научно-технического проекта</i>	Цели и ожидаемые результаты исследования. Трудозатраты и основные функции исполнителей проекта.
3. <i>Планирование научно-исследовательских работ</i>	Составление перечня этапов и работ, составление сметы затрат.
4. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>	Расчёт интегрального показателя эффективности НИР, за счёт определения финансовой и ресурсоэффективности

**Перечень графического материала (с точным указанием обязательных чертежей):**

1. <i>Сегментирование рынка</i>
2. <i>Диаграмма Исикавы</i>
3. <i>QuaD-анализ</i>
4. <i>SWOT-анализ</i>
5. <i>Диаграмма Ганта</i>
6. <i>Календарный план научно-исследовательского проекта</i>

**Дата выдачи задания для раздела по линейному графику**

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Профессор ОСГН ШБИП	Сосковец Любовь Ивановна	д.и.н., доцент		

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8ИМ71	Дергачев Антон Олегович		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

<b>Группа</b> 8ИМ71	<b>ФИО</b> Дергачеву Антону Олеговичу
------------------------	--

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение (НОЦ)</b>	<b>ОИТ</b>
<b>ИШИТР</b>	Магистратура	<b>Направление/специальность</b>	09.04.02 Информационные системы и технологии

Тема ВКР:

Разработка модуля интеграции корпоративной базы знаний на платформе XWiki и системы трекинга задач YouTrack.
--

### Исходные данные к разделу «Социальная ответственность»:

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<p>Объект исследования: модуль интеграции XWiki.          Область применения: использование модуля интеграции может осуществляться на вычислительных устройствах с локальной сетью.          Цель исследования: интеграция вики-системы XWiki и YouTrack.</p>
--	---

### Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p><b>1. Правовые и организационные вопросы обеспечения безопасности:</b></p> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<ul style="list-style-type: none"> <li>• Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 01.04.2019).</li> <li>• ГОСТ Р ИСО 6385-2016. Эргономика. Применение эргономических принципов при проектировании производственных систем.</li> <li>• ГОСТ 12.1.005-88 с изм. №1 от 2000 г., СанПиН 2.2.4.548-96. Общие санитарно-гигиенические требования к воздуху рабочей зоны.</li> </ul>
<p><b>2. Производственная безопасность:</b></p> <p>2.1. Анализ выявленных вредных и опасных факторов</p> <p>2.2. Обоснование мероприятий по снижению воздействия</p>	<ul style="list-style-type: none"> <li>• Отклонение показателей микроклимата;</li> <li>• Превышение уровня шума;</li> <li>• Отсутствие или недостаток естественного света;</li> <li>• Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.</li> </ul>
<p><b>3. Экологическая безопасность:</b></p>	<p>Влияние утилизации:</p> <ul style="list-style-type: none"> <li>• Твердых отходов (системный блок компьютера, принтеры, сканеры, клавиатура, манипулятор «мышь») на литосферу;</li> <li>• Жидкие отходы (сточные воды) на гидросферу;</li> <li>• Люминесцентные лампы.</li> </ul>
<p><b>4. Безопасность в чрезвычайных ситуациях:</b></p>	<p>Выбор типичной ЧС – пожар.</p> <ul style="list-style-type: none"> <li>• разработка превентивных мер по предупреждению ЧС;</li> <li>• разработка действий в результате возникшей ЧС и мер по ликвидации её последствий.</li> </ul>
Дата выдачи задания для раздела по линейному графику	

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ООД ШБИП	Атепаева Наталья Александровна			

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ИМ71	Дергачев Антон Олегович		

## Реферат

Отчет по выпускной квалификационной работе состоит из 122с., 27 рис., 31 табл., 28 источников, 5 приложений.

Ключевые слова: модуль, проектирование, разработка, веб, XWiki, YouTrack, интеграция.

Объектом исследования является разрабатываемая программная система для взаимодействия с корпоративной базой знаний на платформе XWiki и системой трекинга задач YouTrack.

Цель работы – создание модуля расширения для платформы XWiki.

В процессе исследования проводился анализ требований к модулю расширения вики-системы XWiki, анализ архитектур и инструментов для реализации программного продукта.

В результате исследования был спроектирован и разработан модуль расширения к вики-системе XWiki.

Степень внедрения: в данный момент модуль работает, и находится на этапе тестирования. Область применения: процесс разработки программного обеспечения в ОАО ТомскНИПИнефть.

Экономическая эффективность/значимость работы: обусловлена важностью экономии времени в процессе разработки программного обеспечения.

Функциональность модуля может быть расширена добавлением настроек для интеграции с сервером. Так же в дальнейшем планируется опубликовать результат работы в открытый доступ на GitHub и Opensource российского сообщества разработчиков.

## Определения, обозначения, сокращения, нормативные ссылки

ПП – программный продукт

API – Application Programming Interface

ИС - Информационная система.

UML – Unified modeling language

JSON – JavaScript object notation

Модуль расширения - это отдельная часть программного кода, фрагмент программы, который полностью самостоятельно выполняет свою задачу

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	17
1 ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТНОВКА ЗАДАЧИ .....	19
1.1 Анализ литературы используемой в работе .....	19
1.2 Анализ проблем при интеграции информационных систем .....	20
1.3 Анализ существующих методов интеграции информационных систем .....	23
1.3.1 Компоненты информационной системы .....	23
1.3.2 Интеграция платформ.....	25
1.3.3 Интеграция данных .....	26
1.3.4 Интеграция приложений .....	28
1.3.5 Интеграция бизнес-процессов .....	31
1.4 Обзор платформы XWiki.....	32
1.5 Цели работы и задачи.....	33
2 ПРОЕКТИРОВАНИЕ МОДУЛЯ ИНТЕГРАЦИИ.....	35
2.1 Проектирование схемы взаимодействия систем .....	35
2.2 Архитектура модуля расширения .....	35
2.3 Жизненный цикл модуля расширения .....	36
2.4 Построение функциональной диаграммы с декомпозицией.....	37
2.5 Разработка спецификации требований к системе.....	39
2.6 Архитектура формата JSON.....	44
3 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА .....	47
3.1 Выбор языка программирования и среды разработки.....	47
3.2 Взаимодействие модуля интеграции с API XWiki .....	50
3.3 Модель данных XWiki .....	53
3.4 Разработанные методы для модуля интеграции .....	55
3.5 Подключение модуля в XWiki .....	57
3.6 Результат разработки.....	57
4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ .....	65

4.1	Описание поставленных задач в разделе .....	65
4.2	Оценка коммерческого и инновационного потенциала проекта .....	66
4.2.1	Потенциальные потребители продукта .....	66
4.2.2	Анализ конкурентных технических решений .....	68
4.2.3	Технология QuaD .....	70
4.2.4	SWOT-анализ .....	71
4.2.5	Диаграмма Исикавы .....	73
4.3	Инициация проекта .....	74
4.3.1	Цели и результат проекта .....	74
4.3.2	Организационная структура проекта .....	76
4.4	Планирование научно-исследовательских работ .....	76
4.4.1	Структура работ в рамках научно-технического исследования .....	76
4.4.2	Определение трудоемкости и графика выполнения работ .....	77
4.6	Вывод по разделу .....	88
5	СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ .....	90
5.1	Правовые и организационные вопросы обеспечения безопасности .....	90
5.2	Производственная безопасность .....	93
5.3	Анализ опасных и вредных производственных факторов .....	94
5.3.1	Отклонение показателей микроклимата .....	94
5.3.2	Превышение уровня шума .....	95
5.4.3	Отсутствие или недостаток естественного света .....	97
5.4.4	Электромагнитное излучение .....	98
5.4	Экологическая безопасность .....	99
5.5	Безопасность в чрезвычайных ситуациях .....	100
5.6	Вывод по разделу .....	101
	ЗАКЛЮЧЕНИЕ .....	102
	СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ .....	103
	ПРИЛОЖЕНИЕ А .....	106
	ПРИЛОЖЕНИЕ Б .....	119
	ПРИЛОЖЕНИЕ В .....	120

ПРИЛОЖЕНИЕ Г .....	121
ПРИЛОЖЕНИЕ Д .....	122



## ВВЕДЕНИЕ

Интеграция информационных систем необходима для сохранения целостности распределенной информации. Интеграция ИС — это процесс установки связей между информационными системами предприятий для получения единого информационного пространства и организации поддержки бизнес-процессов.

ОАО «ТомскНИПИнефть» - ведущий корпоративный научно-проектный институт в области комплексного проектирования для предприятий нефтегазового комплекса. В институте ведется разработка профильного программного обеспечения для предприятий нефтегазового комплекса, где в ходе работы используются различные инструменты.

На сегодняшний день существует огромное количество таких инструментов прямо и косвенных решающих задачу планирования и отслеживания работы над проектом, с разной степенью эффективности, но будущее за продуктами интегрирующие в себя не только инструменты ввода, сортировки, представления и оформления данных, но также содержащие инструменты аналитики, программирования и поиска [1].

В частности спецификация требований в «ТомскНИПИнефть» ведётся в XWiki, а трекинг задач осуществляется с помощью YouTrack.

Поскольку на текущий момент в «ТомскНИПИнефть» XWiki и YouTrack не имеют интеграции друг с другом - существует проблема переноса этих задач из одной системы в другую с минимальными временными затратами и которая является актуальной на сегодняшний день так как требует автоматизации.

Данная работа является частью общего проекта по интеграции систем XWiki и YouTrack, отражающая взаимодействие между XWiki и веб-сервисом. В работе будут использоваться материалы, представленные при проектировании взаимодействия между YouTrack и веб-сервисом, где веб-сервис является общим посредником при интеграции систем проекта.

Целью данной работы является обеспечение интеграции таких бизнес-процессов предприятия, как разработка программного обеспечения и выявление требований к разработке ПО, путем создания модуля расширения предназначенного для передачи и обработки данных от вики-системы XWiki на веб-сервис.

В разделе 1 проведен анализ литературы используемой в работе, анализ проблем при интеграции информационных систем, анализ существующих методов интеграции информационных систем, описана платформа XWiki, поставлены задачи выпускной квалификационной работы.

В разделе 2 проведено проектирование схемы взаимодействия систем, сделан анализ современных средств разработки, описана архитектура модуля расширения XWiki, его жизненный цикл, построена функциональная диаграмма и произведена её декомпозиция, разработана спецификация требований к системе, описана архитектура json.

В разделе 3 описано API вики-системы и способы взаимодействия с ним, представлена модель данных XWiki, разработаны методы модуля расширения, описано подключение модуля в вики-систему XWiki, и представлены результаты разработки.

Раздел 4 посвящен вопросам финансового менеджмента, ресурсоэффективности и ресурсосбережения. Был произведен расчет затрат на разработку и вычислена экономическая эффективность проекта.

В 5 разделе содержится анализ действующих стандартов безопасности труда при разработке и эксплуатации модуля расширения XWiki.

# **1 ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТНОВКА ЗАДАЧИ**

Перед тем как приступить к проектированию и разработке программного продукта, необходимо провести обзор предметной области.

В данной главе будет проведён анализ теоретических и практических работ по теме интеграция информационных систем. Показана актуальность решаемой задачи, проведен анализ используемой литературы, также поставлена цель данной работы и определены основные задачи.

## **1.1 Анализ литературы используемой в работе**

Литературу по основной части работы ВКР можно условно разделить на три группы.

В первую группу можно включить источники [1, 2, 3, 4] из которых была взята информация, для проведения анализа предметной области.

Во вторую группу можно включить источники литературы, в которых описываются способы возможных решений [5 - 13]. Данная группа является наиболее важной для выполнения выпускной квалификационной работы, так как на основе этих информации этих источников можно создать программный продукт, отвечающий всем требованиям разработки. Статьи и публикации из этой группы были в основном взяты из известных и популярных ресурсов посвящённых разработке программных продуктов и их сопровождения.

К третьей группе источников литературы, относятся рекомендации, спецификации и концепции для эффективного решения поставленных задач магистерской диссертации [19 - 28].

Проведённый анализ показал, что выделенные группы наиболее полно раскрывают предметную область и сущность проблемы. Первая группа источников литературы все специфические термины, характеризующие выбранные технологии. Вторая группа описывает общие

понятия и стандарты разработки для модуля расширения вики-системы XWiki, третья дополнит теоретическую часть работы рекомендациями и концепциями разработки для эффективного решения поставленных задач в рамках выпускной квалификационной работы.

## **1.2 Анализ проблем при интеграции информационных систем**

Под интеграцией информации понимается интеграция данных, находящихся во многочисленных системах, и их представление в унифицированном, согласованном и точном виде, которое предназначено для изучения и обработки данных. Такая интеграция предназначена исключительно для конечных пользователей, которым для выполнения поставленных перед ними задач необходимо работать с многочисленными системами.

Необходимым условием осуществления интеграции данных является проведение досконального анализа, во-первых, задействованных систем и данных с целью определения релевантных данных, подлежащих процедурам извлечения и преобразования, а, во-вторых, целевых структур, в которые будут загружаться эти данные. Подготовка отчетности осуществляется с помощью аналитических средств, которые позволяют каждый раз по-новому взглянуть на собранные данные, т.е. помогают создавать информацию, необходимую для принятия решений.

В зависимости от системы данные могут представляться в разных форматах и с различными метками, которые наиболее уместны с точки зрения их использования, и поэтому для того, чтобы данные были пригодны, пользователю придется их соотносить. Таким образом, могут возникнуть дублирующиеся и недостоверные данные, что сказывается на качестве информации.

Проблемы с качеством информации встречаются в отдельных наборах данных – таких, как файлы и базы данных, – например, как результат ошибок при вводе, утери информации и других загрязнений данных. Когда

интеграции подлежат множество источников данных в интегрированных системах баз данных или глобальных информационных Интернет-системах, необходимость в очистке данных существенно возрастает. Это происходит оттого, что источники часто содержат разрозненные данные в различном представлении. Для обеспечения доступа к точным и согласованным данным необходима консолидация различных представлений данных и исключение дублирующейся информации [2].

В базы данных загружают и постоянно обновляют огромные объемы данных из различных источников, поэтому вероятность попадания в них «грязных данных» весьма высока. Более того, интегрированные информационные системы используются для принятия решений, следовательно, необходимо проводить корректировки загруженных данных, так как дублирующаяся или утраченная информация может стать причиной некорректной или неадекватной статистики. Ввиду большого спектра возможных несоответствий в данных и большого объема данных их очистка считается одной из самых крупных проблем в технологии интеграции данных. Вся очистка данных обычно выполняется в отдельной области подготовки данных до выгрузки преобразованных данных конечному пользователю.

Преобразование данных требуется для поддержки любых изменений в структуре, представлении или содержании данных. Эти преобразования становятся необходимы при изменении структуры данных, переходе на новую информационную систему или в случае, когда нужно интегрировать множественные источники данных.

Проблемы, представленные в отдельных источниках, усугубляются в случае интеграции множества источников данных. Каждый источник может содержать загрязненные, а также представленные различным образом, перекрывающие и противоречащие друг другу данные. Причиной этому – обычно независимая разработка, внедрение и поддержка источников, ориентированная на специфические потребности предприятий. В результате

проявляется значительная неоднородность в системах управления данными, моделях данных, планах схем и текущих данных.

Учитывая, что очистка источников данных представляет собой довольно дорогостоящий процесс, предотвращение ввода загрязненных данных является важным шагом в уменьшении проблем, связанных с очисткой данных. Для этого требуется соответствующим образом спроектированные схема базы данных и ограничения целостности, а также приложения для ввода данных. Кроме того, выявление правил очистки данных в процессе проектирования интегрированной информационной системы может дать основания для улучшения ограничений, вызванных существующими схемами.

Основные проблемы при проектировании схемы базы данных – это конфликты наименований и структурные конфликты. Конфликты наименования происходят, когда одно и то же имя используется для различных объектов или различные имена используются для одного и того же объекта. Структурные конфликты проявляются в различных вариациях и связаны с различными представлениями одного и того же объекта в различных источниках. В дополнение к конфликтам на уровне схемы, множество конфликтов встречается только на уровне элемента данных (конфликты данных). Все проблемы отдельных источников могут появляться в различных представлениях в различных источниках (например, дубликаты записей, противоречащие записи и т.д.). Более того, даже когда встречаются одни и те же имена атрибутов и типы данных, тем не менее, представления информации или интерпретации значений данных от источника к источнику могут различаться. Кроме того, информация в источниках может быть представлена в различных уровнях агрегации или относиться к различным периодам.

Основной проблемой интеграции данных из множества источников является выявление перекрывающихся данных, в особенности – соответствующие друг другу данные, относящиеся к одному и тому же

объекту реального окружения. Эту проблему называют также проблемой идентичности объекта, проблемой исключения дублирования. Часто информация только местами избыточна и источники могут дополнять друг друга, обеспечивая более полную информацию об объекте. Конечно же, дублирующаяся информация должна удаляться, но при этом дополняющая информация должна консолидироваться и соединяться, чтобы объекты реального окружения получили согласованное представление.

В заключение можно выделить основные проблемы интеграции источников данных – это конфликты имен и структурные конфликты на уровне схемы базы данных, а на уровне элемента данных можно наблюдать различные представления информации и дублирующиеся записи. Решение данных проблем требует одновременно и интеграции схемы, и очистки данных. Для этого необходимо урегулировать в первую очередь конфликты схемы, чтобы обеспечить возможность очистки данных, в особенности – выявление дубликатов на основе унифицированного представления информации. [3]

### **1.3 Анализ существующих методов интеграции информационных систем**

#### **1.3.1 Компоненты информационной системы**

Информационная система представляет собой совокупность нескольких компонентов, поэтому, говоря об интеграции информационных систем, следует подразумевать интеграцию составляющих их компонентов.

Обычно, информационная система содержит в себе следующие компоненты, представленные на рисунке 1.1:

- Платформа, на которой функционируют остальные компоненты системы, включающая в себя аппаратуру (железо) и системное ПО;
- Данные, с которыми работает система. Состоят из СУБД и баз данных;

- Приложения, реализующие бизнес-логику по работе с данными системой. Состоят из компонентов бизнес-логики, пользовательского интерфейса, вспомогательных компонентов (фреймворк) и сервера приложений, который обеспечивает хранение и доступ к компонентам приложения;
- Бизнес-процессы, представляющие из себя сценарии работы пользователей с системой.

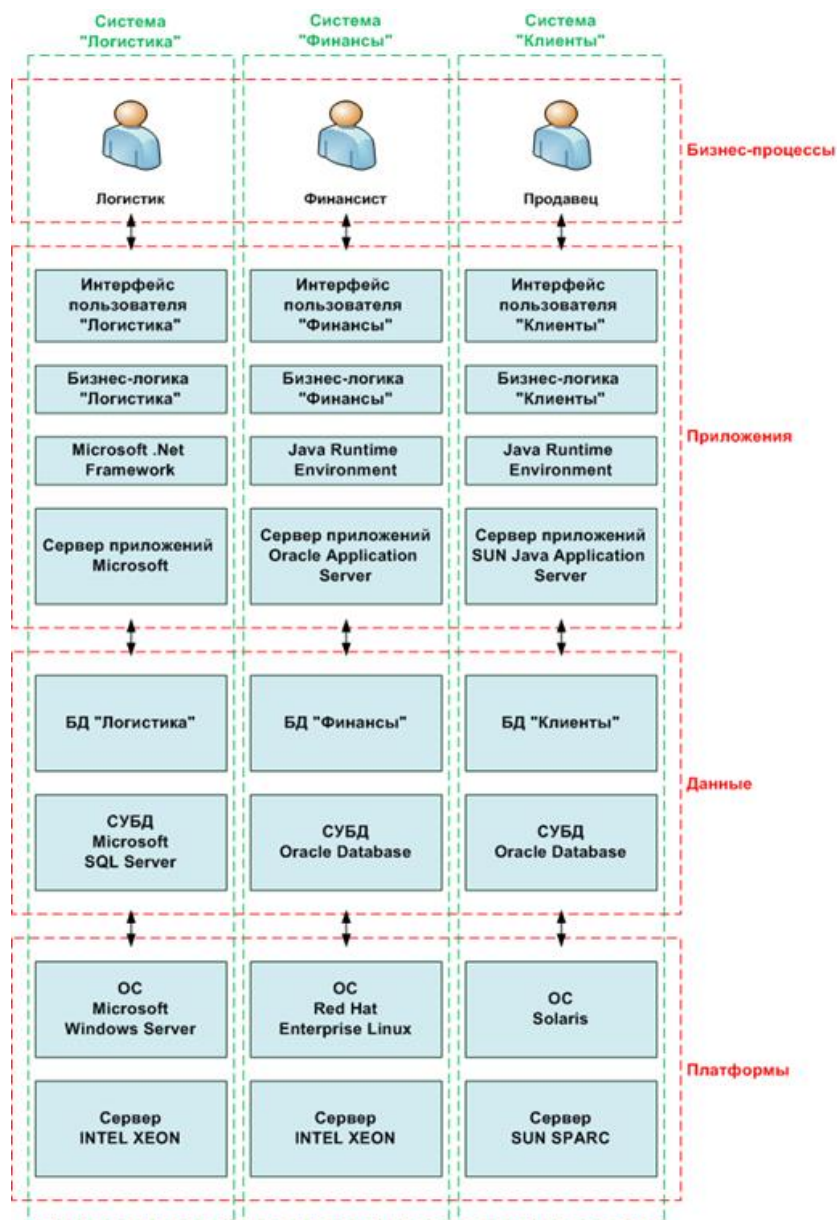


Рисунок 1.1 – Основные компоненты информационных систем

В результате, интеграция информационных систем заключается в интеграции одного или нескольких компонентов интегрируемых информационных систем (объектов интеграции).



### 1.3.2 Интеграция платформ

Целями интеграции платформ являются:

- Обеспечение возможности взаимодействия между приложениями, работающими на различных программно-аппаратных платформах (например, между приложениями, работающими на серверах Windows, Solaris, Linux и др.);
- Обеспечение возможности работы приложений, разработанных для одной программно-аппаратной платформы, на других программно-аппаратных платформах (например, приложений Windows на платформах Linux, Solaris и др.).

Существует несколько подходов, направленных на достижение этих целей. В рамках каждого из подходов существуют различные технологии:

- Удаленный вызов процедур (RPC, Web-сервисы, REST и пр);
- ПО промежуточного слоя (Microsoft.Net, Java Runtime);
- Виртуализация.

Технологии удаленного вызова процедур (в широком смысле под процедурой понимается некоторая функциональность приложения) позволяют опубликовать процедуру и обеспечить возможность ее вызова (передачи входящих параметров и получения выходных результатов) для приложений, работающих на других платформах.

Концепция программного обеспечения промежуточного слоя (framework, среда исполнения, виртуальная машина) состоит в разработке прикладного ПО не с использованием сервисов конкретной операционной системы (например, Windows API), а с использованием сервисов ПО промежуточного слоя. Разработчиками ПО промежуточного слоя создаются ее реализации под различные операционные системы, которые транслируют вызовы соответствующих функций фреймворка в вызовы соответствующей операционной системы. Типичным примером является технология Java Runtime Environment. Приложения, разработанные для этой технологии

работают на любых программно-аппаратных платформах (Windows, Linux и др.) без каких-либо доработок самих приложений. Аналогичные возможности предоставляет среда Microsoft .Net Framework.

Интересной и современной концепцией является «виртуализация». К интеграции платформ она имеет отношение постольку, поскольку позволяет существенно упростить использования различных платформ и, соответственно, использование систем, требующих для своего функционирования наличия конкретных платформ. Если без виртуализации возможно одновременное функционирование  $N$  операционных сред на  $N$  серверов, то применение технологий виртуализации позволяет обеспечить функционирование  $N$  операционных сред на  $M$  серверов. Если  $N > M$  – это позволяет сократить расходы на аппаратное обеспечение путем его более эффективного использования. Если  $N < M$  – это простой путь увеличения производительности систем. Например, виртуализация позволяет развернуть и одновременно использовать на одном физическом сервере несколько операционных систем: Windows, Linux и др. На каждом из таких «виртуальных» серверов могут быть развернуты соответствующие системы, которые будут доступны одновременно. Примеры технологий виртуализации: Microsoft Hyper-V, KVM, OpenVZ, Virtuozzo, VMware, Xen и пр.

### **1.3.3 Интеграция данных**

По определению информационная система работает с данными. В подавляющем большинстве случаев система имеет в своем составе базу данных для их хранения. Интеграция на уровне данных предполагает совместное использования данных различных систем. Интеграция данных может оказаться проще, чем интеграция приложений, т.к. промышленные СУБД, в которых обычно хранят данные информационные системы, имеют развитые возможности программного доступа к данным из других приложений. Сами приложения при этом могут иметь весьма ограниченные

возможности программного (вне собственного пользовательского интерфейса) использования своей функциональности внешними системами.

Подходы к интеграции данных:

- Универсальный доступ к данным;
- Хранилища данных.

Технологии универсального доступа к данным позволяют обеспечить единообразный доступ к данным различных СУБД. Посредником для работы с конкретной СУБД в данном случае является драйвер для соответствующей СУБД. Например, один и тот же SQL-запрос на выборку данных `SELECT * FROM TTABLE` может быть использован на выборку данных из таблицы `TTABLE`, хранящейся в различных СУБД. Это позволяет абстрагироваться от специфики конкретных СУБД и легко осуществлять интеграцию данных, хранящихся в различных СУБД. Наиболее распространенные технологии этого класса: ODBC, JDBC, ADO.NET. Кроме того, на сегодняшний день широко распространены технологии объектно-реляционного отображения (ORM), которые также позволяют абстрагироваться от деталей взаимодействия с конкретными СУБД. Примерами таких технологий являются Entity Framework, Hibernate, NHibernate, Flexberry ORM.

Концепция хранилищ данных состоит в создании корпоративного хранилища данных. Хранилище данных – база данных, хранящая в себе данные, собираемые из баз данных различных информационных систем, для целей их дальнейшего анализа. Например, может быть создано единое хранилище данных компании, в которое собрана информация из бухгалтерской, оперативной системы, внешних систем партнеров компаний. Для создания хранилищ данных используются технологии (OLAP), отличные от технологий создания оперативных БД (OLTP). В основном это делается для повышения производительности выполнения сложных аналитических запросов по многим параметрам (многомерные запросы). Подходы к созданию и наполнению хранилищ данных отражены в парадигме ETL (extraction, transformation, loading = извлечение, преобразование и загрузка).

Технологии и инструментальные средства анализа больших массивов данных с целью выявления закономерностей предметной области объединяются понятием «Data Mining». Термин для совокупности технологий хранилищ данных и инструментальных средств, обеспечивающих перевод транзакционной деловой информации в человекочитаемую форму, пригодную для бизнес-анализа – «Business Intelligence».

### **1.3.4 Интеграция приложений**

Интеграция на уровне приложений подразумевает использование готовых функций приложений другими приложениями. Например, разрабатывая систему электронного документооборота, существует возможность использовать в рамках этой системы в качестве текстового редактора MS Word вместо того, чтобы разрабатывать свой собственный текстовый редактор. Или, например, ПО Call-центра, получив входящий звонок от клиента, имеет возможность обратиться к функции биллинговой системы по проверке баланса (на входе – номер телефона абонента, на выходе – его текущий баланс) и, в зависимости от состояния баланса соединить его с оператором или автоматически проинформировать о необходимости пополнить свой счет. При этом структура база данных биллинговой системы является ее внутренней информацией, публикуются конкретные функции, позволяющие другим системам работать с конкретными данными.

Стоит упомянуть следующие подходы к интеграции приложений:

- Интерфейсы прикладного программирования;
- Обмен сообщениями (Корпоративная сервисная шина);
- Сервис-ориентированная архитектура;
- Интеграция пользовательских интерфейсов.

Программный интерфейс конкретной системы представляет собой «опубликованный» функционал этой системы, который может быть

использован извне. Функционал может публиковаться в виде набора функций (пример – Windows API) или в виде объектной модели (объекты со свойствами и методами, пример – объектные модели приложений Microsoft Office).

В большинстве случаев интеграция нескольких систем заключается в передаче информации между ними, например, в форме запрос-ответ. Если системы функционируют в гетерогенных распределенных средах, то принципиальное значение имеет обеспечение гарантированности, безопасности, управляемости доставки информации между приложениями. Эти и другие принципы реализуются в корпоративных системах обмена сообщениями. В данном случае речь идет об обмене сообщениями между приложениями, а не людьми, как, например, в случае E-mail или мессенджеров. Функциональность этих систем достаточно прозрачна – прием сообщения от одного приложения, транспортировка по заданным правилам и передача этого сообщения другому приложению. При этом может производиться шифрование сообщений (для невозможности прочтения данных в процессе транспортировки), цифровая подпись (для защиты от умышленного изменения данных во время пути сообщения), настройка подписки (для отправки одного сообщения сразу нескольким приложениям), определение метаданных для сообщений (для облегчения использования сообщений со сложной структурой содержимого) и др.

Сервис-ориентированная архитектура (SOA) является логическим продолжением концепции Веб-сервисов, которая состоит в публикации функциональных блоков какого-либо приложения в виде, позволяющем получить к ним доступ другим приложениям через Веб. Веб (протокол HTTP) в данном случае привлекателен ввиду возможности его использования и, соответственно, использования опубликованного в Веб-приложений функционала на любых программно-аппаратных платформах. Веб-сервис – небольшая программная надстройка над функционалом приложения, преобразующая вызовы, получаемые через Веб, во внутренние вызовы

функций приложения и возвращающая результаты обратно. Основными идеями SOA являются:

- Публикация функционала корпоративных приложений в виде Веб-сервисов. Упорядочивание опубликованных сервисов в виде каталога;
- Построение на основе Веб-сервисов новых приложений путем их комбинации.

Цена создания новых приложений на основе существующих Веб-сервисов будет существенно ниже, чем разработка приложений «с нуля» или обширная интеграция с другими системами.

Например, в компании (оператор связи) существует система Service Desk (техническая поддержка абонентов) и биллинговая система (тарификация услуг). Перед компанией стоит задача сделать новую систему «Личный кабинет абонента», в которой абонент мог бы через Интернет просмотреть состояние своего счета и сообщить о неисправности. Для этого компания вместо того, чтобы создавать «Личный кабинет» с собственной базой данных, синхронизируемой с БД биллинговой системы и системы Service Desk, использует готовые Web-сервисы «Карточка абонента» (опубликованный функционал биллинговой системы) и «Создать заявку в техподдержку» (опубликованный функционал системы Service Desk). Очевидно, что вся работа по новому приложению «Личный кабинет» состоит лишь в создании Веб-интерфейса пользователя на сайте компании.

Также часто используется следующий подход – интеграция пользовательских интерфейсов. Например, для создания приложений «одного окна». Простейший пример – фреймы на Веб-странице. Внутри каждого фрейма содержится отдельное Веб-приложение. Благодаря фреймам, все эти приложения отображаются на экране одновременно. Пользовательские интерфейсы Веб-приложений легко интегрируются, однако, существуют возможности интегрировать и «классические» пользовательские интерфейсы и их фрагменты (ActiveX).

### 1.3.5 Интеграция бизнес-процессов

Наиболее целостным подходом к интеграции систем является интеграция на уровне бизнес-процессов. В рамках интеграции бизнес-процессов происходит и интеграция приложений, и интеграция данных и, что не менее важно, людей, вовлеченных в этот бизнес-процесс. Интеграция на уровне бизнес-процессов является наиболее «естественной» для организаций, так как их деятельность состоит, прежде всего, именно из бизнес-процессов, а не приложений, баз данных и платформ.

Идеи, лежащие в основе интеграции бизнес-процессов, достаточно просты:

- Составить сценарий некоторого бизнес-процесса, происходящего в организации, описать в нем операции взаимодействия пользователей с различными системами и систем между собой. Таким образом, бизнес-процесс является элементом, логически интегрирующим различные системы. Сценарий создается при помощи специализированного программного продукта, который далее будет управлять ходом этого бизнес-процесса согласно сценарию.

- Операции взаимодействия с системами в рамках бизнес-процесса детально описываются в терминах информационного обмена: форматы обмена, используемые сервисы, приложения, события, правила, политики и т.п.

- К интегрирующему программному обеспечению, при помощи которого описан сценарий бизнес-процесса, подключаются посредством адаптеров интегрируемые системы, вовлеченные в бизнес-процесс. Таким образом, становится возможным автоматизированный информационный обмен между системами.

- Готовый к выполнению бизнес-процесс выводится на «пульт управления» менеджера, при помощи которого, он может запускать и останавливать бизнес-процессы, отслеживать их состояние, вводить данные и

принимать решения на отдельных операциях бизнес-процессов, требующих участия человека и др. Взаимодействия между системами, не требующее участия человека осуществляется автоматически интегрирующим ПО.

## 1.4 Обзор платформы XWiki

XWiki - это бесплатная open-source платформа, написанная на Java. XWiki корпоративная вики, она включает в себя инструмент для редактирования WYSIWYG, импорт/экспорт документов на основе OpenDocument, а также расширенное управление дополнениями.

XWiki позволяет создавать мини-приложения на вики-страницах (рисунок 1). В XWiki интегрирована система управления страницами, система контроля версий, система прав доступа. Так же XWiki умеет экспортировать страниц в PDF, и имеет горячие клавиши [4].

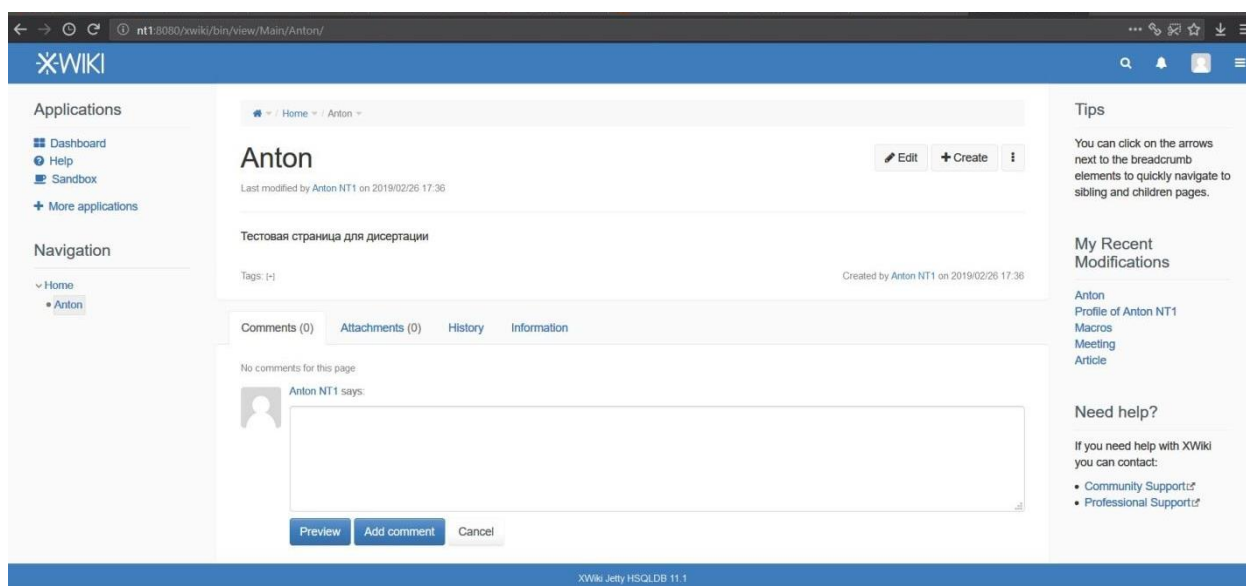


Рисунок 1.1 – Интерфейс статьи в XWiki

Как вики-приложение, XWiki позволяет хранить структурированные данные и выполнять сценарий на стороне сервера в интерфейсе вики. Языки сценариев, включая Velocity, Apache Groovy, Python, Ruby и PHP, могут быть записаны непосредственно в вики-страницы с помощью макросов вики[4].

Созданные пользователем структуры данных могут быть определены в вики-документах, а экземпляры этих структур могут быть присоединены к



вики-документам, сохранены в базе данных и опрошены с использованием либо языка запросов Hibernate, либо собственного языка запросов XWiki [5].

XWiki - приложение-вики, которое позволяет создавать объекты и классы внутри вики. Таким образом, формы могут быть разработаны за очень короткий промежуток времени и могут использоваться повторно для ввода данных в вики, следуя определенному шаблону. Это означает, что конечным пользователям может быть предоставлена страница, на которой уже нарисован макет, где они могут напрямую заполнить необходимые поля [6].

Исходный код XWiki распространяется по лицензии GNU Lesser General Public License и размещается на GitHub, где каждый может свободно размещать исходный код и вносить изменения в свой собственный репозиторий [7].

## **1.5 Цели работы и задачи**

Целью данной выпускной квалификационной работы является разработка программного модуля предназначенного для передачи и обработки данных от вики системы XWiki на веб-сервис по средствам своего API.

Для достижения поставленной цели, требуется решить следующие задачи:

- Выявить требования для разработки модуля интеграции;
- Выявить требования вики системы XWiki к модулям расширения своего функционала;
- Спроектировать модуль интеграции для вики системы XWiki;
- Разработать спроектированный модуль вики системы XWiki.

В результате выполненных работ в рамках данного раздела магистерской диссертации был проведен анализ литературы и используемой в данной работе, проведен анализ проблем при интеграции информационных

систем, описаны существующие методы интеграции информационных систем, проведен обзор вики системы XWiki, указаны цели и задачи выполняемой работы.

## 2 ПРОЕКТИРОВАНИЕ МОДУЛЯ ИНТЕГРАЦИИ

После того как проведен анализ и обоснована необходимость выполнения данной работы, необходимо произвести проектирование программного продукта для его последующей реализации.

### 2.1 Проектирование схемы взаимодействия систем

Подключение модуля XWiki происходит посредством API XWiki и менеджера расширений. Взаимодействие модуля с веб-сервисом осуществляется посредством API модуля и API веб-сервиса. Данные передаются модулем в формате JSON на веб-сервис, где они анализируются и передаются на YouTrack посредством REST API.

Общая схема интеграции представлена на рисунке 2.1.

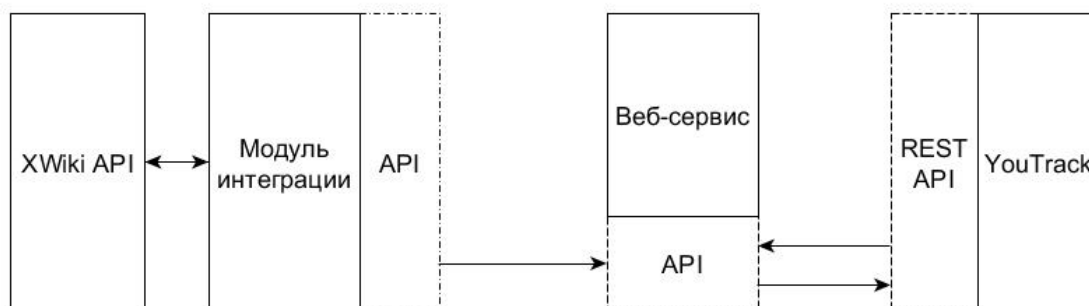


Рисунок 2.1 – Схема интеграции XWiki и YouTrack

Передача данных модулем осуществляется по средством своего API в формате JSON.

### 2.2 Архитектура модуля расширения

Разработка модулей для вики системы XWiki должна следовать концепции созданной самими разработчиками XWiki, а именно: архитектура модуля должна состоять из двух частей:

- Модуль должен реализовать интерфейс XWikiPluginInterface. Так же он может расширять класс XWikiDefaultPlugin, который является адаптером к XWikiPluginInterface. Сам модуль содержит основные функции.

- API модуля должно расширять класс Api. Само API будет содержать все открытые методы.

### 2.3 Жизненный цикл модуля расширения

Когда платформа XWiki инициализирована, менеджер модулей вызывает конструктор класса для всех включенных модулей (классы, реализующие `com.xpn.xwiki.плагин.XWikiPluginInterface`), жизненный цикл модуля интеграции представлен на рисунке 2.2.



Рисунок 2.2 – Жизненный цикл модуля интеграции

Для каждого модуля конструктор класса вызывается только один раз, и менеджер модулей вызывает метод `init(XWikiContext)` модуля, рисунок 2.3.



Рисунок 2.3 – Инициализация работы модуля интеграции в XWiki

Каждый раз, когда происходит обращение к модулю, при вызове метода, обслуживаемого API модуля `set ( integrationYouTrack = «xwiki.integrationYouTrack.getPage()»)` XWiki вызывает метод `getPluginApi ()` для экземпляра модуля, который сам создает экземпляр класса `com.xpn.xwiki.PluginApi`.

Метод `getPluginApi()` создает новый экземпляр класса `PluginApi` каждый раз, когда необходимо получить доступ к API для модуля. Один экземпляр модуля охватывает весь жизненный цикл сервлета.

## 2.4. Построение функциональной диаграммы с декомпозицией

Спроектированная система предназначена для анализа и передачи данных на веб-сервис введенных пользователем в вики систему XWiki.

Система имеет следующие функции:

- 1) Анализ выбранных данных;
- 2) Формирование и отправка JSON;
- 3) Создание, удаление страницы требований.

На основе функций системы был смоделирован процесс интеграции данных при помощи нотации функционального моделирования IDEF0, диаграмма IDEF0 представлена на рисунке 2.4.

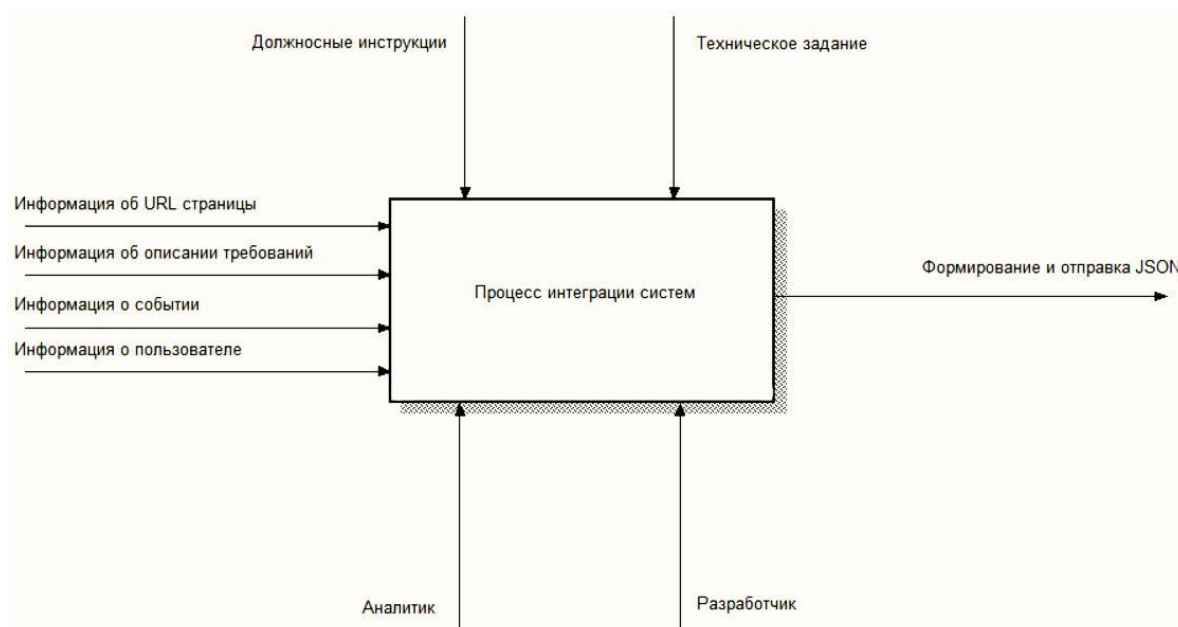


Рисунок 2.4 – Диаграмма IDEF0 информационной системы

Весь процесс интеграции данных можно декомпозировать на подсистемы. Декомпозиция процесса интеграции данных представлена на рисунке 2.5.

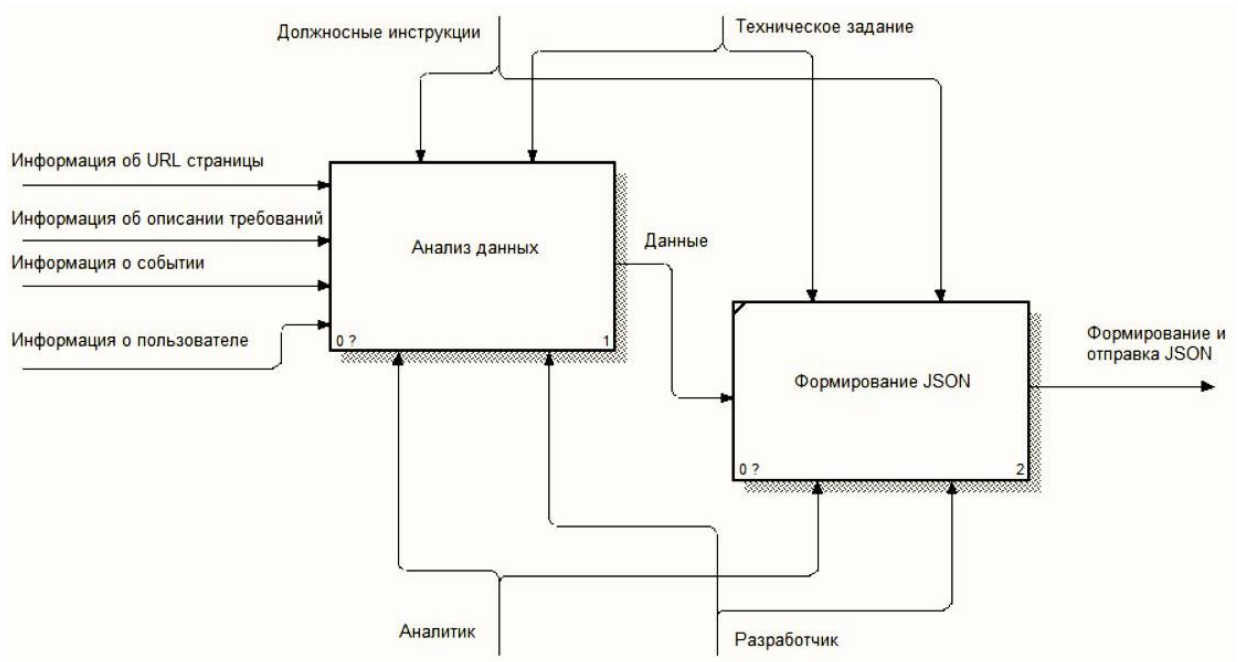


Рисунок 2.5 – Декомпозиция процесса интеграции данных

Создание функциональной модели подсистемы «Анализ данных» показывает, какие информационные преобразования выполняются в данной операции бизнес процесса, функциональная диаграмма представлена на рисунке 2.6.

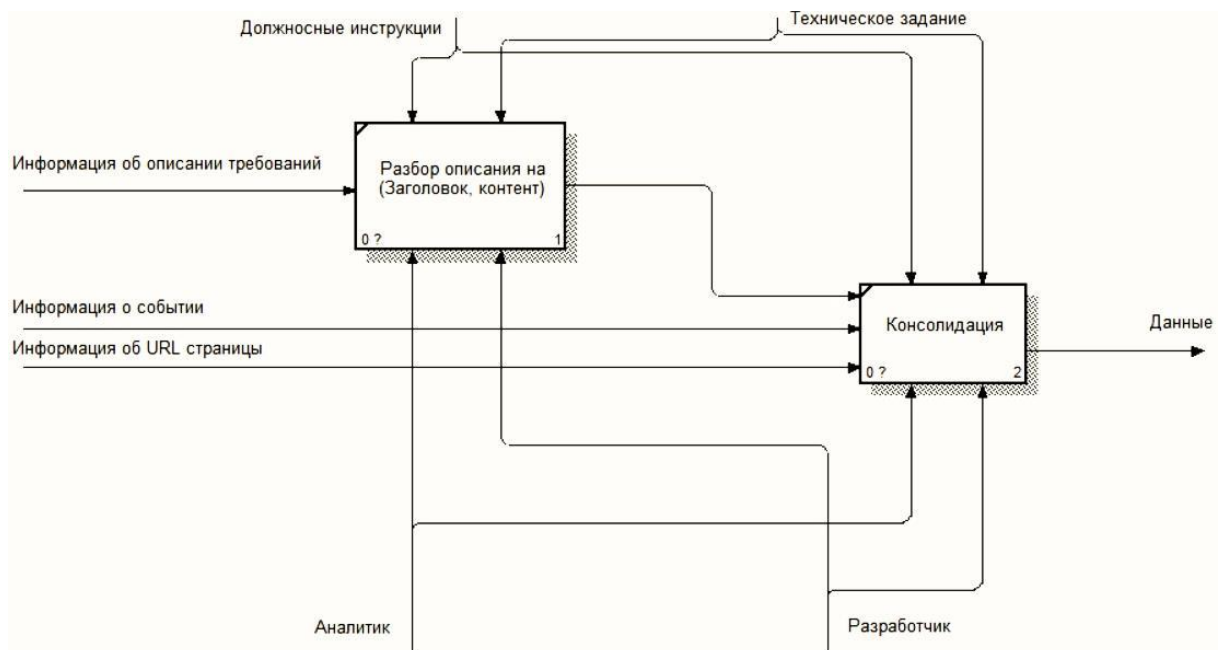


Рисунок 2.6 – Функциональная диаграмма операции «Анализ данных»

Функция разбора описания позволяет разбить содержимое выбранной текстовой области пользователя, на «заголовок» и «контент» для последующего создания дочерней страницы в вики системе XWiki, и упаковки в JSON для отправки на веб-сервис.

Консолидация - объединяет результаты прошлых преобразования и передаёт для дальнейшей обработки.

## 2.5. Разработка спецификации требований к системе

После того как была спроектирована архитектура модуля согласно концепции по разработке модулей для вики системы XWiki от разработчиком данной системы и спроектирован жизненный цикл модуля интеграции, следует разработка требований к разрабатываемому модулю интеграции.

Таблица 2.1 - Требования к системе

Символ	Группы требований
F	Общие функциональные требования
I	Требования к интерфейсу пользователя
P	Требование средствам интеграции
IS	Требование к информационной безопасности
TS	Требование к техническому обеспечению
SR	Требование к программному обеспечению
RG	Требование к надежности
D	Требования к документации

Таблица 2.2 - Общие функциональные требования

Код требования	Требование	Примечание
F.01	Требования к обработки данных	
F.01.01	Система должна обеспечивать преобразование данных из вики-системы в формат JSON	
F. 01.02	Система должна обеспечивать преобразование данных из формата JSON	
F,02	Требования к передачи данных	
F.02.01	Система должна обеспечивать получение данных из вики-системы XWiki	
F.02.02	Система должна обеспечивать передачу данных по средствам своего API	
F.02.03	Система должна обеспечивать отправку данных в формате JSON на веб-сервис	

Таблица 2.3 - Требование к интерфейсу пользователя

Код требования	Требование	Примечание
I.01	Система должна обеспечивать наличие кнопок «Создать требование» «Обновить требование» «Удалить требование»	

Таблица 2.4 - Требование средствам интеграции

Код требования	Требование	Примечание
<b>P.01</b>	<b>Требования к интеграции с XWiki</b>	
P.01.01	Система должно обеспечивать интеграцию с вики-системой XWiki для передачи данных	



Продолжение таблицы 2.4

P.01.02	Система должно обеспечивать взаимодействие с системой XWiki по средством менеджера расширений XWiki	
<b>P.02</b>	<b>Требования к интеграции с веб-сервисом</b>	
P.02.01	Система должно обеспечивать интеграцию с веб-сервисом, по средствам API веб-сервиса	
P.02.02	Система должно обеспечивать обмен данными по средствам API веб-сервиса	

Таблица 2.5 - Требование к информационной безопасности

Код требования	Требование	Примечание
IS.01	Система должна поддерживать проведение комплекса мероприятий по обеспечению информационной безопасности	Защиты от несанкционированного доступа, нарушения целостности и достоверности хранящейся и передаваемой информации
IS.02	В системе должна обеспечиваться конфиденциальность данных	

Таблица 2.6 - Требования к техническому обеспечению

Код требования	Требования	Примечания
TS.01	Требования для Web-сервера	
TS.01.01	4 ядерный процессор 3,2 ГГц	
TS.01.02	Оперативная память: 8 Gb	
TS.01.03	Свободное дисковое пространство 2 Gb	
TS.01.04	Канал связи между серверами: 100 Мбит/с	
TS.01.05	Внешний канал связи: 100 Мбит/с	
TS.02	Требования для сервера базы данных	
TS.02.01	4 ядерный процессор 3,2 ГГц	
TS.02.02	Оперативная память: 8 Gb	
TS.02.03	Свободное дисковое пространство 2Gb	
TS.02.04	Канал связи между серверами: 100 Мбит/с	

Таблица 2.7 - Требования к программному обеспечению

Код требования	Требования	Примечания
SR.01	Требования к Операционным системам	
SR.01.01	Система должна работать на ОС Microsoft Windows, Linux, Apple MacOS	
SR.02	Требования к браузеру	

Продолжение таблицы 2.7

SR.02.01	Internet Explorer 9 и выше (только для Windows); Mozilla Firefox 13 и выше; Safari 5 и выше; Google Chrome 15 и выше; Opera 11 и выше.	
SR.03	Требования к базе данных	
SR.03.01	База данных MySQL	
SR.03.02	Система управления базой данных	

Таблица 2.8 - Требования к надежности

Код требования	Требования	Примечания
RG.01	Программное обеспечение системы должно поддерживать восстановление своего функционирования при перезапуске аппаратных средств.	
RG.02	Система должна обеспечить функционирования под управлением сервера базы данных с поддержкой мультиплатформенности, многопроцессорности.	
RG.03	Система должна обеспечить повышенный уровень сохранности информации при авариях, отказах технических средств (в том числе – потеря питания)	

Продолжение таблицы 2.8

RG.04	Система должна поддерживать устойчивость по отношению к программно-аппаратным ошибкам	
RG.05	Система должна обеспечивать сохранность и целостность ранее введенных данных при обновлениях	

Таблица 2.9 - Требования к документации

Код требования	Требования	Примечания
D.01	Разработка документации на API системы должна осуществляться в соответствии с настоящим Техническим требованием	
D.02	Документация должна быть оформлена в текстовом формате на русском языке и представлена Заказчику в электронном виде в 2-х экземплярах	

## 2.6. Архитектура формата JSON

JSON (JavaScript Object Notation) - формат обмена данными. JSON - текстовый формат, полностью независимый от языка реализации, но он использует соглашения, знакомые программистам C-подобных языков, таких как C, C++, C#, Java, JavaScript, Perl, Python и многих других. Эти свойства делают JSON идеальным языком обмена данными.

JSON представляет данные в двух структурах:

- Коллекция пар ключ/значение. В разных языках, эта концепция реализована как объект, запись, структура, словарь, хэш, именованный список или ассоциативный массив.
- Упорядоченный список значений. В большинстве языков это реализовано как массив, вектор, список или последовательность.

Почти все современные языки программирования поддерживают их в какой-либо форме.

Схемы нотаций JSON:

Объект - неупорядоченный набор пар ключ/значение. Объект начинается с { (открывающей фигурной скобки) и заканчивается } (закрывающей фигурной скобкой). Каждое имя сопровождается : (двоеточием), пары ключ/значение разделяются , (запятой), рисунок 2.7.

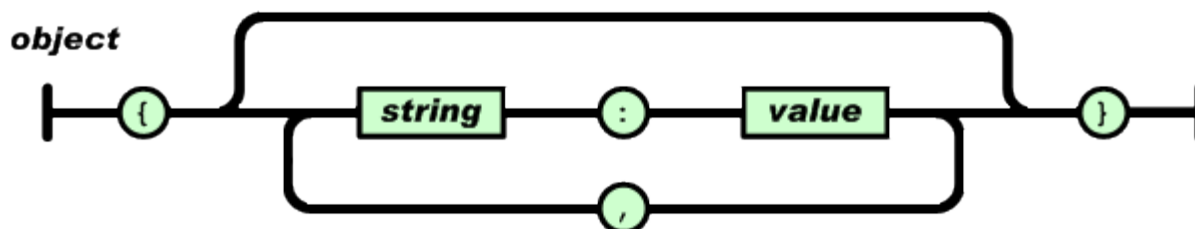


Рисунок 2.7 – Схема объекта в формате JSON.

Массив - упорядоченная коллекция значений. Массив начинается с [ (открывающей квадратной скобки) и заканчивается ] (закрывающей квадратной скобкой). Значения разделены , (запятой), рисунок 2.8.

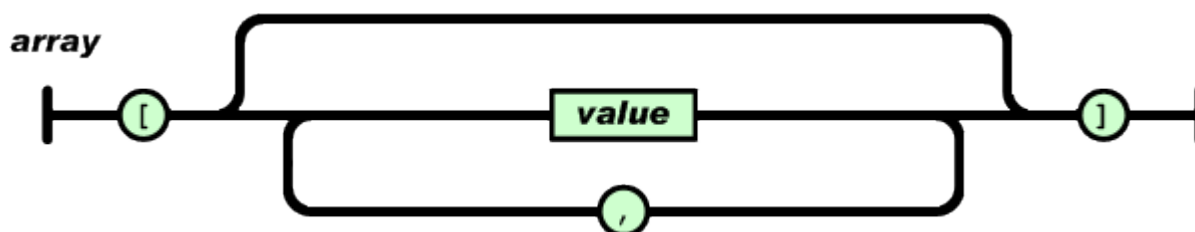


Рисунок 2.8 – Схема массива в формате JSON.

Значение может быть строкой в двойных кавычках, числом, true, false, null, объектом или массивом. Эти структуры могут быть вложенными, рисунок 2.9.

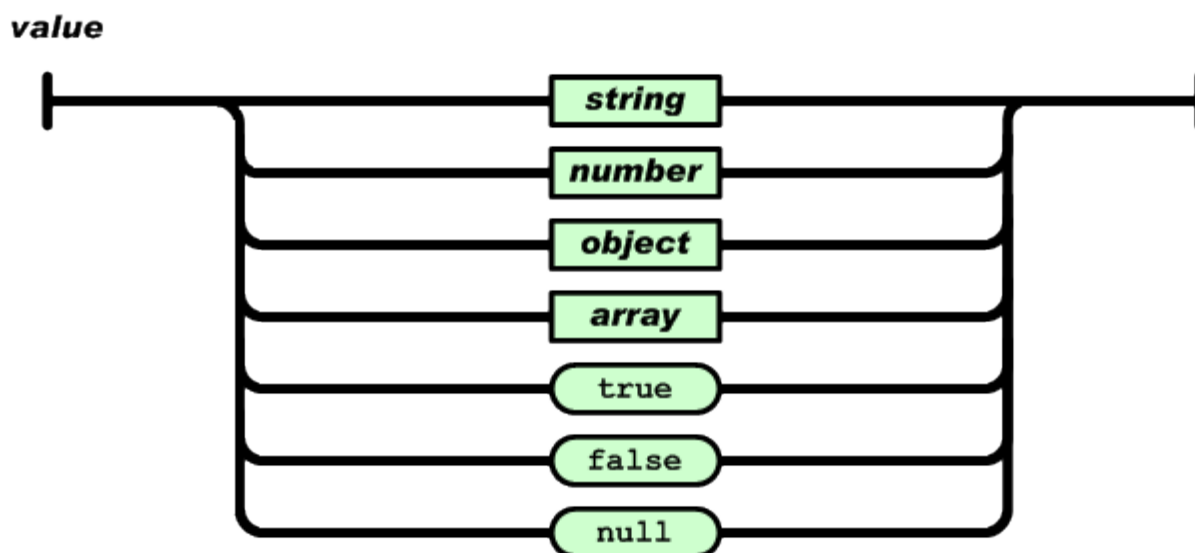


Рисунок 2.9 – Схема упорядоченного списка в формате JSON.

Элементы могут быть вложены друг в друга как на стороне браузера, так и на стороне сервера. На практике формат JSON (описание стандарта RFC 4627) предоставляет значительно более 4 уровней вложенности.

JSON принято относить к конструкциям данных, которые проще XML, понятны одновременно и людям, и компьютерам.

В результате выполненных работ в рамках данного раздела выпускной квалификационной работы проведен анализ современных средств разработки и выбран язык программирования и интегрированная среда разработки. Спроектирована архитектура модуля, его жизненный цикл, построены функциональные диаграммы с декомпозицией, разработана спецификация требований к системе и описана архитектура формата JSON.

### **3 РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА**

После того как произведен анализ и проектирование модуля интеграции вики-системы XWiki, необходимо произвести разработку программного продукта для его последующего внедрения.

#### **3.1 Выбор языка программирования и среды разработки**

Вики система XWiki поддерживает расширения разработанные на языке программирования Java. Для решения поставленных магистерской диссертации, был выбран язык программирования Java.

Большинство крупных компаний используют Java в своих разработках. Многие серверные приложения, обрабатывающие десятки миллионов запросов, написаны на Java. Java – универсальный язык программирования, его используют компании разного масштаба в своем корпоративном серверном ПО. Этот язык работает на всех платформах, не смотря на постоянные обновления и модернизацию, старый Java-код совместим с новым.

Существует множество областей применения Java, от сайтов электронной коммерции до Android приложений, от научных до финансовых приложений, таких как трейдинговые системы, от игр, типа Minecraft, до настольных программных средств, таких как Eclipse, Netbeans и IntelliJ, от open source фреймворков до J2ME приложений и т.д.

IntelliJ IDEA — интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains. Программа содержит полный набор необходимых для создания полноценных приложений компонент: редактор, среда компиляции и выполнения, а также отладчик.

JetBrains IntelliJ IDEA – ведущая среда быстрой разработки на языке Java. IntelliJ IDEA представляет собой комплекс тесно интегрированных

инструментов программирования, включающий интеллектуальный редактор исходных текстов, инструменты рефакторинга кода, встроенную поддержку технологий J2EE, механизмы интеграции со средой тестирования Ant/Junit/Maven и системами управления версиями, а также визуальный конструктор графических интерфейсов [12]. Сравнение JetBrains IntelliJ IDEA с другими средами разработки рассмотрено в таблице 1.

Таблица 3.1 - Сравнения IDE

IDE	Платформы	Языки	Система контроля версий	Сообщество	Цена
IntelliJ IDEA	Windows, Linux, macOS	Java, JavaScript, HTML, CSS, Python, Ruby, SQL, Groovy	Есть	Большое	\$499 – годовая подписка, для физлиц. Для студентов подписка бесплатная.
NetBeans	Windows, Linux, macOS	C/C++/Fortran, Java, JavaScript, HTML, CSS, Python, Ruby, SQL, Groovy	Есть	Большое	Free
Eclipse	Windows, Linux, macOS	Perl, C++, Java, JavaScript, HTML, CSS, Python, Ruby, SQL	Есть	Большое	Free

Особенности IntelliJ IDEA в том, что данная среда разработки понимает контекст, индексирует весь проект и строит синтаксическое дерево, что существенно ускоряет процесс написания кода и позволяет избежать ошибок.

Maven - это инструмент для сборки проекта: компиляции, создания jar и создания дистрибутива программы. Инструмент сборки проекта - система сборки, для сбора других программ. На вход система сборки получает



исходный код, а на выход выдаёт программу, которую уже можно запустить[8].

Основные преимущества Maven:

- Независимость от ОС. Сборка проекта происходит в любой операционной системе. Файл проекта один и тот же.
- Управление зависимостями. Редко какие проекты пишутся без использования сторонних библиотек (зависимостей). Эти сторонние библиотеки зачастую тоже в свою очередь используют библиотеки разных версий. Maven позволяет управлять такими сложными зависимостями. Что позволяет разрешать конфликты версий и в случае необходимости легко переходить на новые версии библиотек.
- Возможна сборка из командной строки. Такое часто необходимо для автоматической сборки проекта на сервере (Continuous Integration).
- Хорошая интеграция со средами разработки. Основные среды разработки на java легко открывают проекты которые собираются с помощью maven. При этом зачастую проект настраивать не нужно - он сразу готов к дальнейшей разработке.
- Как следствие - если с проектом работают в разных средах разработки, то maven удобный способ хранения настроек. Настроечный файл среды разработки и для сборки один и тот же - меньше дублирования данных и соответственно ошибок[9].

Maven — является одним из самых популярных сборщиков для Java, он так же помогает решать сложные задачи по управлению проектом [10].

Идея Maven - работа над обработкой проекта делится на определенный набор этапов — в Maven это называется «фаза». Под каждую фазу (редактирование, компилирование, сбор) существует уже готовая программа — плагин (plugin). Каждый плагин обладает рядом настроек, которые можно менять под свои нужды [11].

### 3.2 Взаимодействие модуля интеграции с API XWiki

Вики система XWiki обеспечивает доступ ко всем элементам через API (программный интерфейс приложения), основанного на семантики HTTP, RESTful API[18].

Доступ к сервису по умолчанию осуществляется через точку доступа «<http://localhost:8080/xwiki/rest>». Так ресурсы /wikis на сервере localhost на порту 8080, можно получить по следующему URL-адресу:

«<http://localhost:8080/xwiki/rest/wikis>».

Данный способ получение ресурсов позволяет получить их в формате XML.

Помимо извлечения контента в формате XML, также можно извлечь его в формате JSON:

«<http://localhost:8080/xwiki/rest/wikis/xwiki/spaces/Main/pages/WebHome?media=json>». – результат запроса представлен на рисунке 3.1.

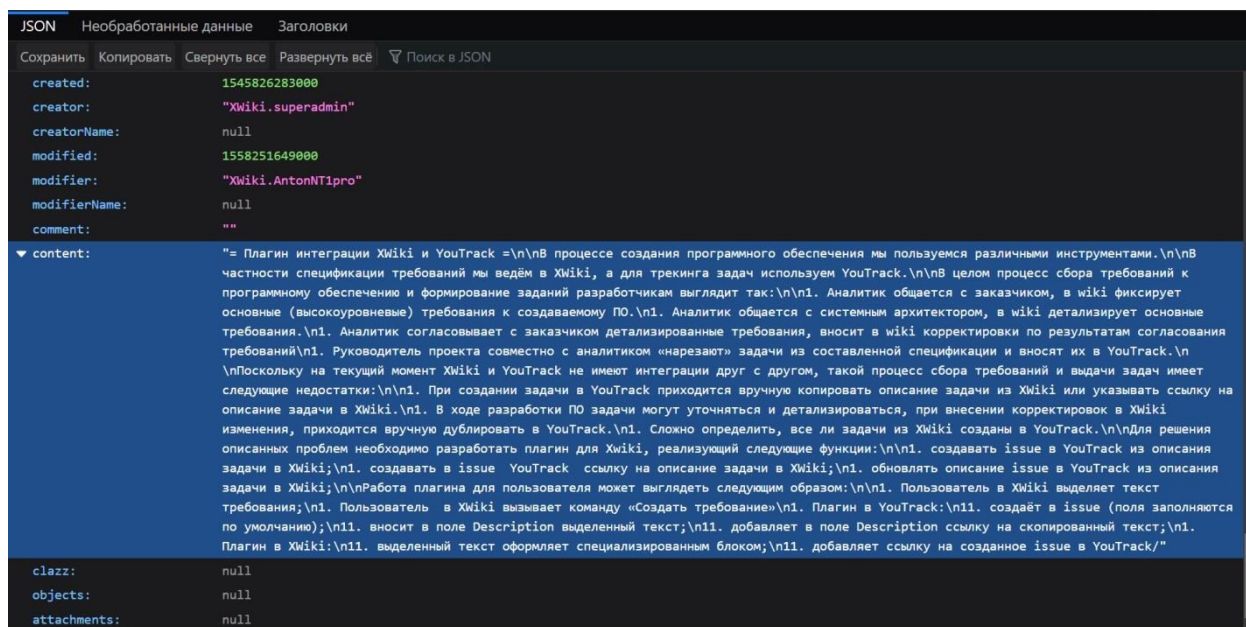
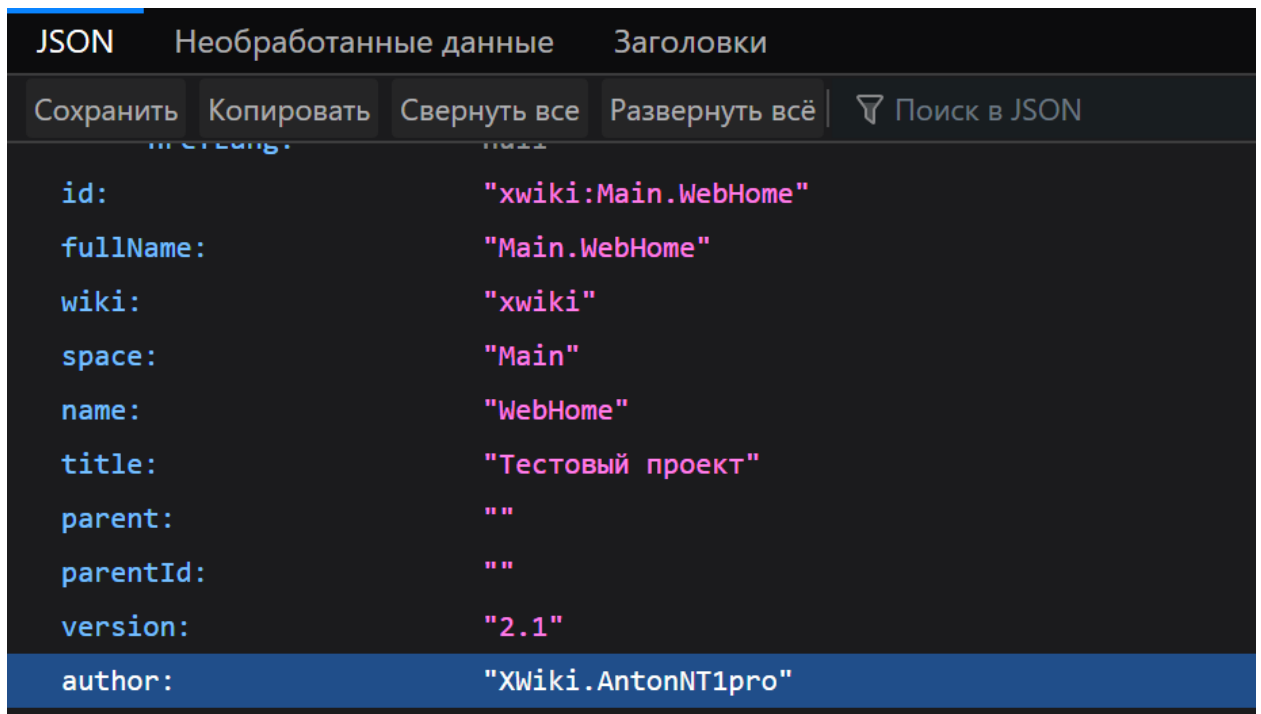


Рисунок 3.1 – Результат запроса получение контента в формате JSON.

Используя RESTful API можно обращаться к объектам страниц и их свойствам, на рисунке 3.2. представлен результат запроса получения автора статьи.

A screenshot of a JSON viewer interface. At the top, there are tabs for 'JSON', 'Необработанные данные', and 'Заголовки'. Below the tabs are buttons for 'Сохранить', 'Копировать', 'Свернуть все', 'Развернуть всё', and a search icon with the text 'Поиск в JSON'. The main area displays a JSON object with the following key-value pairs: 'id' with value 'xwiki:Main.WebHome', 'fullName' with 'Main.WebHome', 'wiki' with 'xwiki', 'space' with 'Main', 'name' with 'WebHome', 'title' with 'Тестовый проект', 'parent' with an empty string, 'parentId' with an empty string, 'version' with '2.1', and 'author' with 'XWiki.AntonNT1pro'. The 'author' line is highlighted in blue.

```
id: "xwiki:Main.WebHome"
fullName: "Main.WebHome"
wiki: "xwiki"
space: "Main"
name: "WebHome"
title: "Тестовый проект"
parent: ""
parentId: ""
version: "2.1"
author: "XWiki.AntonNT1pro"
```

Рисунок 3.2 – Результат запроса получения автора статьи в формате JSON.

Страницы могут иметь один или несколько объектов. Объекты являются экземплярами класса, который содержит набор свойств. Некоторые объекты могут быть непосредственно представлены как объекты первого класса, такие как комментарии и теги [14].

Важной концепцией в REST является наличие ресурсов (источников информации), на каждый из которых имеется глобальный идентификатор (URI в HTTP) [15]. Для манипулирования этими ресурсами используются компоненты сети (пользовательские агенты), и серверы приложений осуществляющие связь через стандартизированный интерфейс (HTTP) и обмениваются представлениями этих ресурсов, схема ресурсов страниц XWiki представлена на рисунке 3.2.

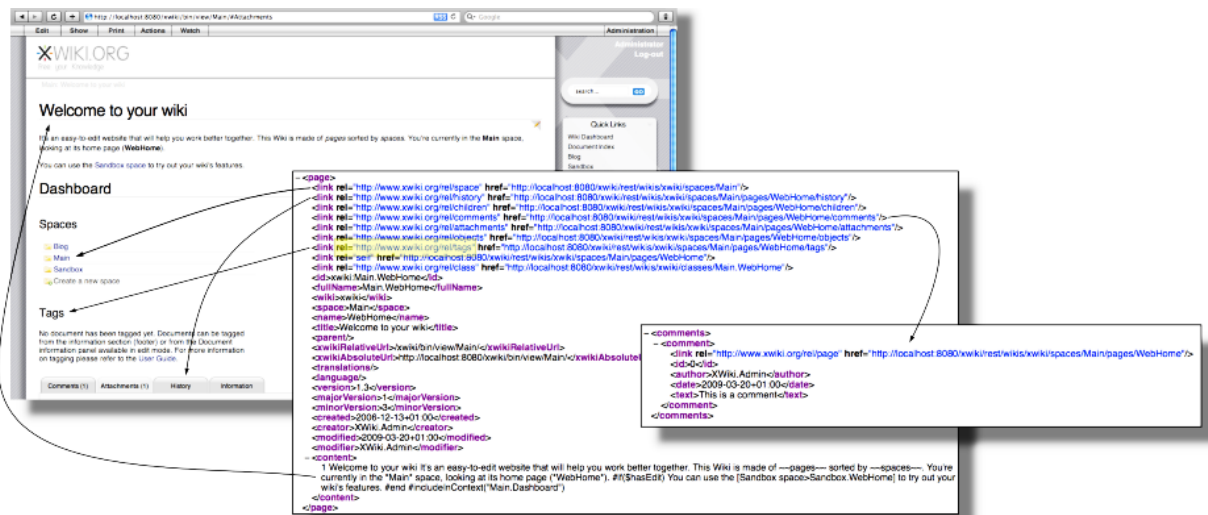


Рисунок 3.3 - Схема ресурсов страниц XWiki

Взаимодействие с XWiki RESTful API, данный интерфейс доступен через HTTP, и для его использования можно использовать любой клиент способный работать по HTTP, для взаимодействия с RESTful API так же может использоваться веб-браузер.

API-интерфейс XWiki RESTful поддерживает два типа аутентификации:

- HTTP BASIC Auth - предоставление своих учетные данные, используя HTTP.
- Сеанс XWiki - используются куки-файлы, предоставляемые механизмом аутентификации, также будет пройдена аутентификация в XWiki RESTful API.

При разработке модуля интеграции вики системы XWiki в рамках выполнения преддипломной практики был использован тип аутентификации HTTP BASIC Auth.

При отправке данных на веб-сервис происходит генерация уникального ключа. Ключ генерируются на основе данных предназначенных для передачи. Эти данные шифруются библиотекой msrpyt по сгенерированному ключу и инициализирующему вектору.

Инициализирующий вектор представляет собой блок случайных данных сгенерированных библиотекой `mscorlib` и предназначен для уникальности шифра при каждой передаче данных. В полученной зашифрованной строке происходит попарная перестановка символов для увеличения степени защищенности передаваемых данных. Преобразованная строка кодируется библиотекой `base64` для передачи по протоколу HTTP. Полученный в результате шифр является ключом для валидации передаваемых данных на веб-сервисе. Данный способ шифрования является двусторонним, подлежит шифрованию и дешифрованию.

Для взаимодействия с RESTful API в Java проект были импортированы ресурсы XWiki:

```
import org.xwiki.component.annotation.Component;
import org.xwiki.rest.XWikiResource;
```

### **3.3 Модель данных XWiki**

Для того что бы пользоваться доступными и открытыми ресурсами XWiki посредством RESTful API стоит рассмотреть модель данных вики системы XWiki [13], рисунок 3.4.

Фактически существует почти взаимно однозначное соответствие между открытыми ресурсами и сущностями этой модели.

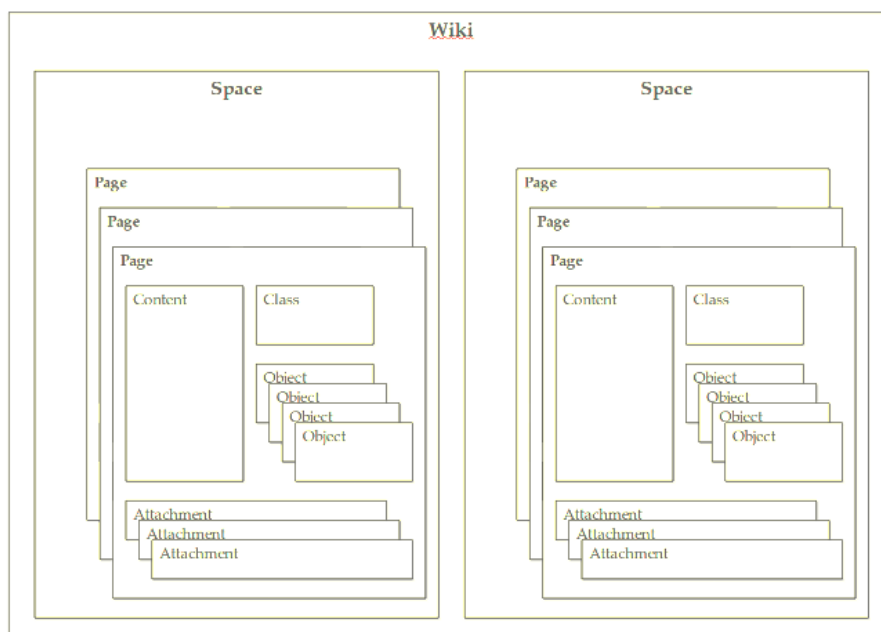


Рисунок 3.4 – Схема модели данных вики системы XWiki

Страница - центральная сущность. Страница имеет связанное содержимое и набор метаданных, предоставляющих дополнительную информацию (например, название, автор, дата создания, дата изменения и т.д.). Содержимое страницы обычно пишется с помощью разметки XWiki и может содержать сценарии, которые могут обращаться к базовой платформе для получения и отображения данных, хранящихся в системе.

Со страницей можно связать три вида сущностей:

- Вложения - это двоичные объекты, связанные со страницей.
- Классы - это метаданные, определяющие структуру. Класс предоставляет определение нескольких свойств. Каждое свойство имеет тип и несколько атрибутов, которые используются для ограничения значений, которые могут быть назначены этому свойству, и способа его редактирования.
- Объекты представляют структурированные данные по схеме, определенной классом. С помощью объектов можно структурировать данные и получить к ним доступ.

Классы и объекты - делают XWiki мощной и более гибкой платформой при сравнении с прочими аналогами. Данные, хранящиеся с

помощью объектов, можно редактировать, искать, манипулировать и отображать определенным образом; это связано с их четко определенной структурой. Приложения могут использовать этот механизм для определения, хранения и редактирования доменных данных с помощью платформы XWiki.

Полная карта ресурсов, доступных через API RESTful представлена в приложение А,Б, на схеме показано, как с точки входа можно перемещаться по всем ресурсам, предоставляемым API RESTful [17].

Узлы - шаблоны URI, задающие адрес набора ресурсов. Шаблон URI-это параметрический URI, задающий структуру для адресации коллекции ресурсов. Ребра-это связи между представлениями различных ресурсов.

Полная схема базы данных представлена в приложение В и Г.

### **3.4 Разработанные методы для модуля интеграции**

Методы:

- `getRangeText(int: x1,x2)` – принимает позиции букв начала и конца в выделенном тексте, возвращает выбранный текст используя Range объект, и передаёт методу `getConstruct(String: inStr)`.
- `createPage(String: header, content)` – метод создания страницы, вызывает метод RESTful API XWiki для создания страницы, принимает результат выполнения метода `getConstruct()`, «header» и «content».
- `getPage()` – метод получения данных страницы XWiki. Данный метод задействует RESTful API XWiki «`xwiki/rest/wikis/xwiki/spaces/Main/pages/WebHome?media=json`» для получения данных страницы в формате JSON. Производит его разбор и передает значение ключа «content» методу `getConstruct(String: str)`.
- `getConstruct(String: str)` – метод разбивает полученную текстовую строку на две части, «header» и «content», далее вызывает метод

getHeaderIndex(), конкатинирует результат работы метода с «header», и передаёт значения методу sendJSON().

- getUsername() – метод для получения данных пользователя, данный метод задействует RESTful API XWiki для получение данных пользователя в формате JSON, производит его разбор и передаёт значение ключа «User» методу sendJSON().

- getKey(String: headerStr, contentStr) – метод производит конкатенацию полученных данных «header», + случайное сгенерированное число от 100 до 1000 (getHeaderIndex() ), и «content», и шифрует полученную строку. Полученный ключ используется в методе sendJSON().

- deletePage(int: idPage) - метод удаления страницы требования, метод осуществляет отправку данных в формате JSON на веб-сервис с указанием типа операции «удаление».

- sendJSON(String: header, content, user, key, url) – метод осуществляет отправку данных в формате JSON, код данного метода:

```
{
HttpPost request = new HttpPost("http://10.192.92.17/api/post/create");
StringEntity params =new
StringEntity("details={\"neader\":header,\"content\":content,\"user\":user,
\"key\":key, \"url\":url,}");
request.addHeader("content-type", "application/x-www-form-urlencoded");
request.setEntity(params);
HttpResponse response = httpClient.execute(request);
}
```

«Header» – содержит заголовок станицы с требованием.

«Content» – содержит описательную часть требования.

«User» – содержит имя автора статьи с требованием.

«Key» – содержит зашифрованную строку.

«Url» – содержит ссылку станицы с требованием.



При создание страницы с требованием, запрос обращается к API веб-сервиса по адресу «<http://localhost/api/post/create>».

При обновление страницы требований, запрос обращается к API веб-сервиса по адресу «<http://localhost /api/post/update>».

При удаление страницы требований, происходит запрос к API веб-сервиса по адресу «<http://localhost /api/post/delete>».

### **3.5 Подключение модуля в XWiki**

Интеграция модуля в вики систему XWiki может осуществляться двумя способами:

- Копирование разработанных jar файлов в каталог предназначенного для расширений вики системы XWiki, Регистрация модуля осуществляется его записью в файле `xwiki.cfg` расположенного в каталоге WEB-INF.

- Использовать систему развертывания расширений вики системы XWiki по адресу [extensions.xwiki.org](http://extensions.xwiki.org), для публикации расширения требуется регистрация, описательная часть разработанного модуля и скриншоты его работы [16].

В рамках выполнения данной магистерской работы расширение в вики систему XWiki было интегрировано путем копирования разработанных файлов в каталог XWiki.

### **3.6 Результат разработки**

Результатом раздела разработка программного продукта – является разработанный модуль интеграции для вики системы XWiki. Модуль позволяет создавать страницы требований и отправлять выбранные данные пользователем на веб-сервис о создание, обновление, удаление, страниц требований в проекте XWiki. Алгоритм работы представлен на рисунке 3.5, на рисунках 3.6 – 3.13 представлена работа модуля.

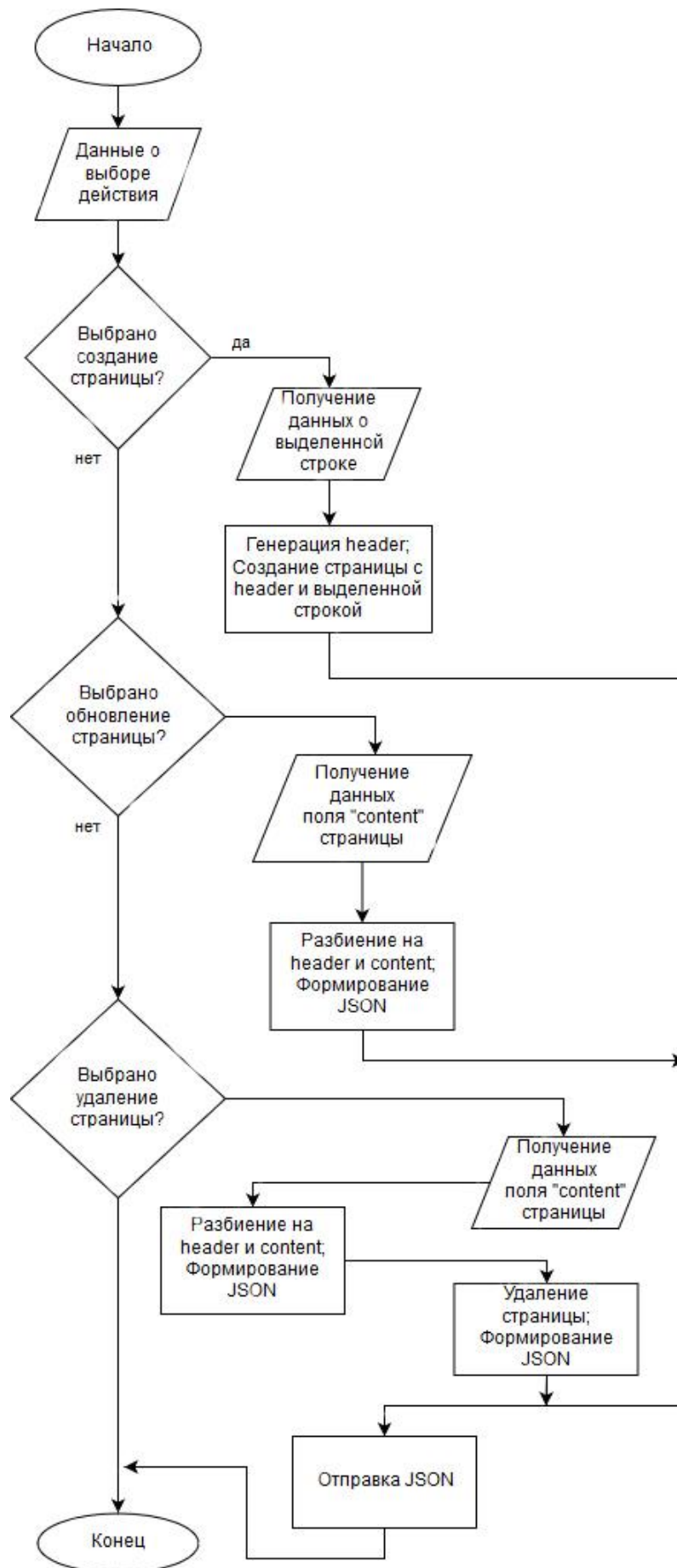


Рисунок 3.5 – Алгоритм передачи данных на веб-сервис

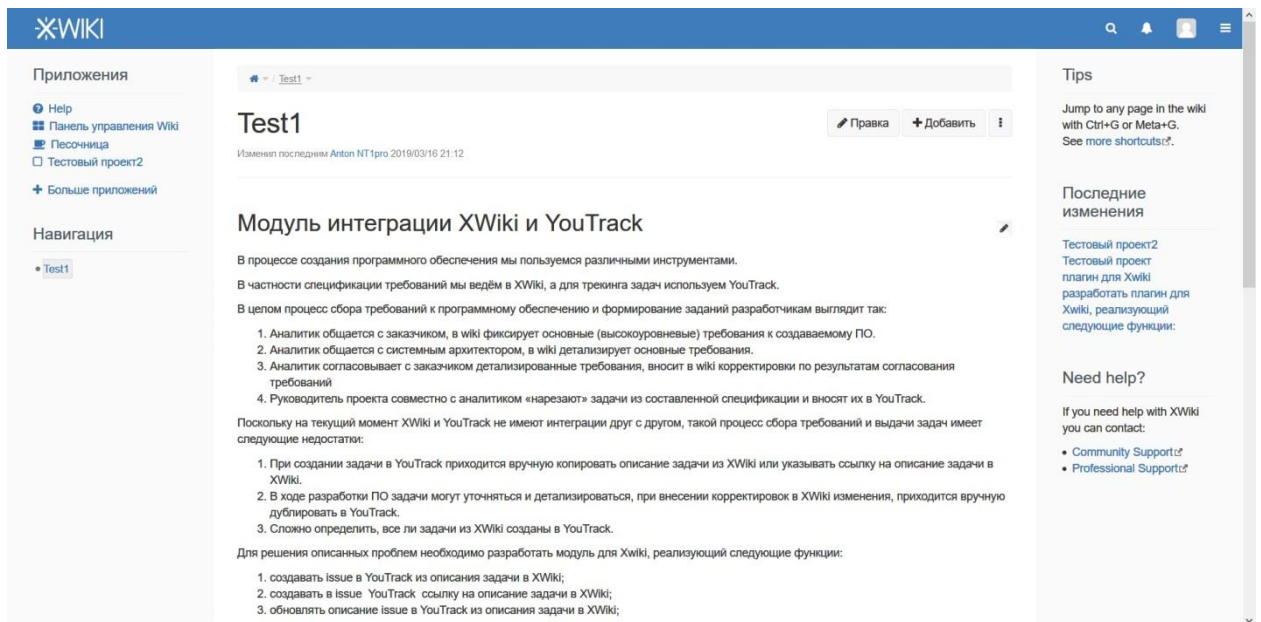


Рисунок 3.6 – Главная страница проекта XWiki

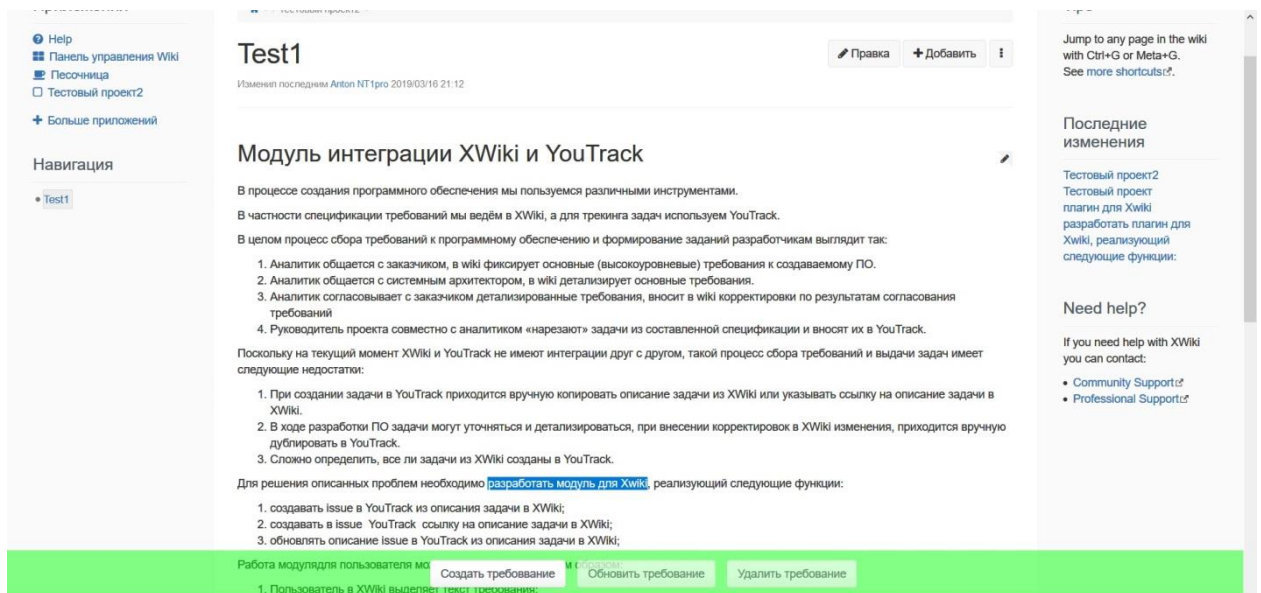


Рисунок 3.7 – Всплывающая панель действий

Всплывающая панель действий, становится доступной пользователю при выделении текста на странице требований проекта и позволяет создать страницу требований.

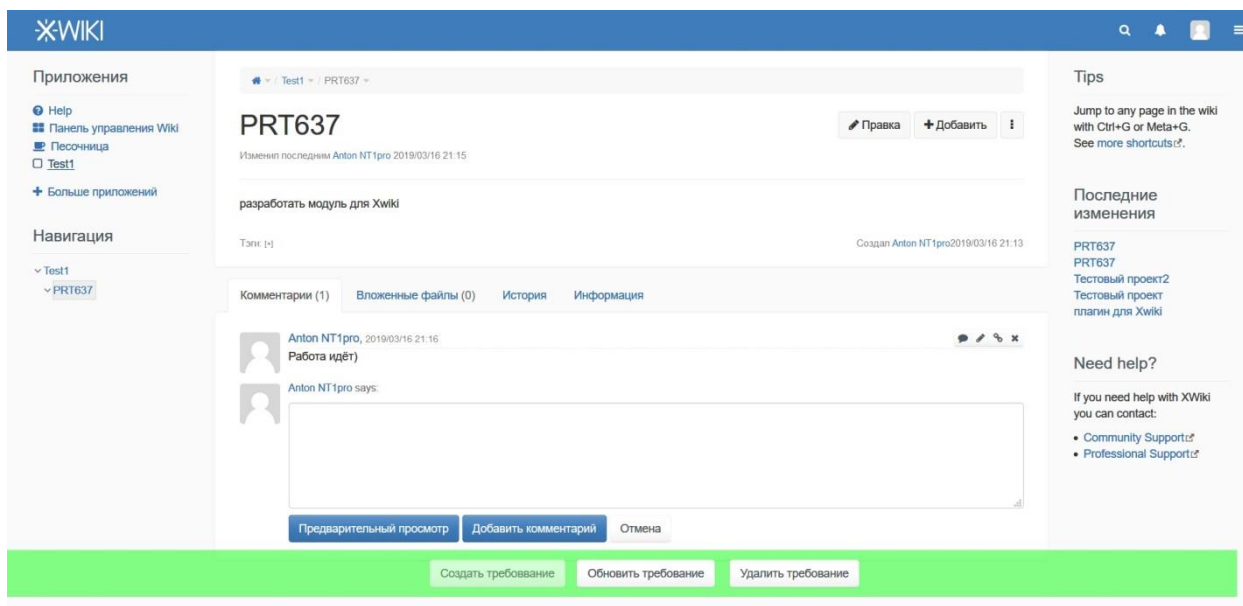


Рисунок 3.8 – Созданная страница с требованием.

При создание старинцы требования был отправлен JSON на веб-сервис:

```
{
  "header": "PRT637",
  "content": "разработать модуль для Xwiki",
  "url": "https://localhost:8080/xwiki/bin/view/Test1/PRT637",
  "user": "AntonNT1pro",
  "key":
  "UFJUNjM30YDQsNC30YDQsNCx0L7RgtCw0YLRjCDQvNC+0LTRg9
  C70Ywg0LTQu9GPIFh3aWtp/aHR0cHM6Ly9udDE6ODA4MC94d2lraS9
  iaW4vdmllldy9UZXRN0MS9QUlQ2Mzc="
}
```

Всплывающая панель доступна пользователю на созданной странице требования, и позволяет обновить требование (отправить обновление требования на веб-сервис), удалить страницу требования (отправить запрос на удаление страницы).

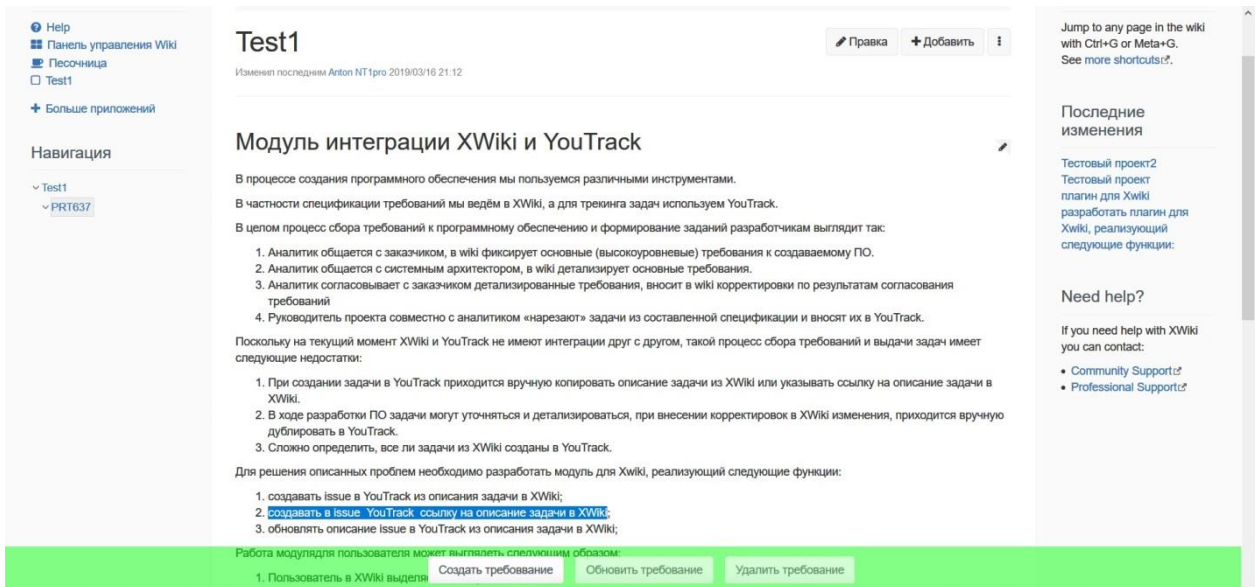


Рисунок 3.9 – Выделение текста для создания страницы с требованием

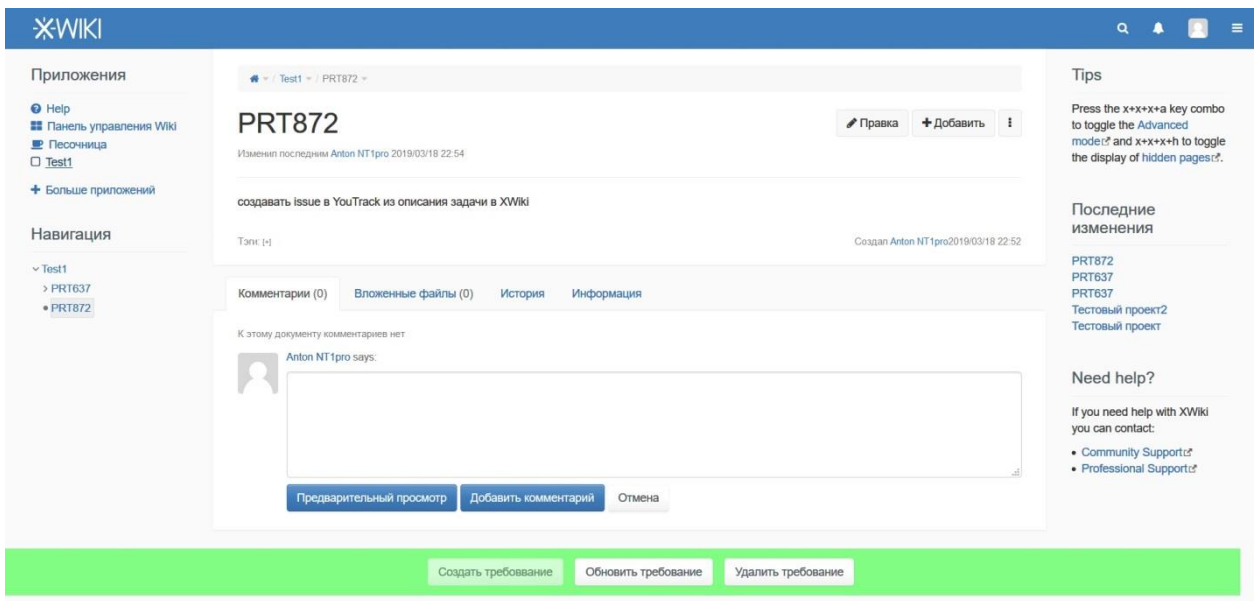


Рисунок 3.10 – Страница созданного требования из предыдущей страницы требований.

При создание страницы требования был отправлен JSON на веб-сервис:

```
{
  "header": "PRT872",
  "content": "создавать issue в YouTrack из описания задачи в XWiki",
  "url": "https://localhost:8080/xwiki/bin/view/Test1/PRT872",
  "user": "AntonNT1pro",
```



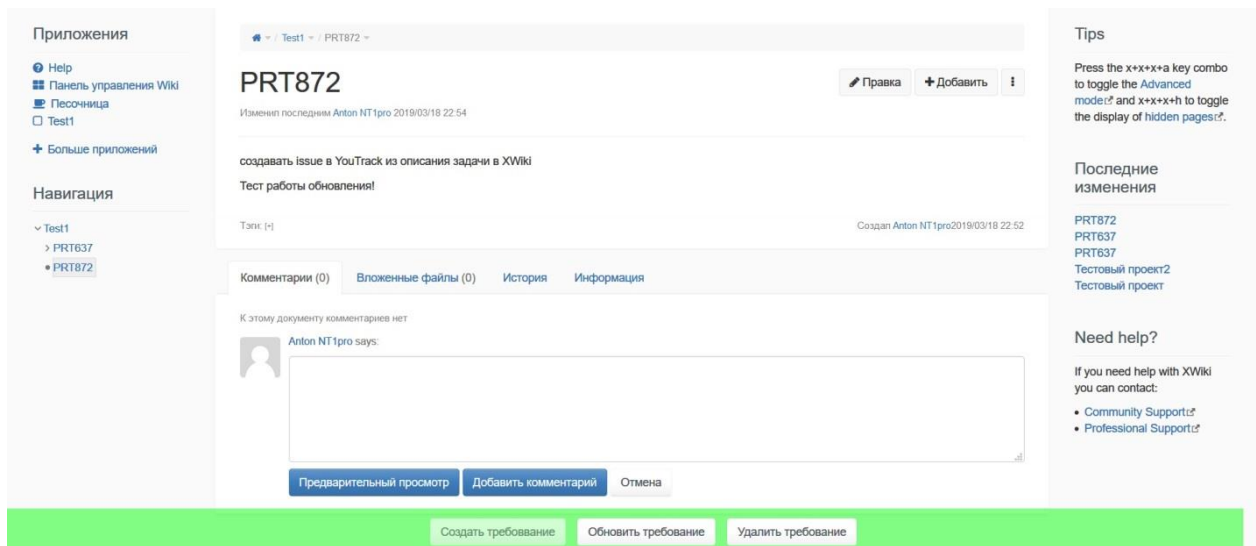


Рисунок 3.12 – Обновление требования на страницы

При обновлении требования был отправлен JSON на веб-сервис:

```
{
  "header": "PRT872",
  "content": "создавать issue в YouTrack из описания задачи в XWiki
  Тест работы обновления!",
  "url": "https://localhost:8080/xwiki/bin/view/Test1/PRT872",
  "user": "AntonNT1pro",
  "key": "UFJUODcy0YHQvtC30LTQsNCy0LDRgtGMIGlzc3VIINC
  yIFlvdVRyYWNrINC40Lcg0L7Qv9C40YHQsNC90LjRjyDQt9Cw0LTQs
  NGH0Lgg0LIgWFdpa2kg0KLQtdGB0YIlg0YDQsNCx0L7RgtGLINC+0L
  HQvdC+0LLQu9C10L3QuNGPIQ==/aHR0cHM6Ly9udDE6ODA4MC94
  d2lraS9iaW4vdmllZy9UZXRN0MS9QUIQ4NzI=" }
```



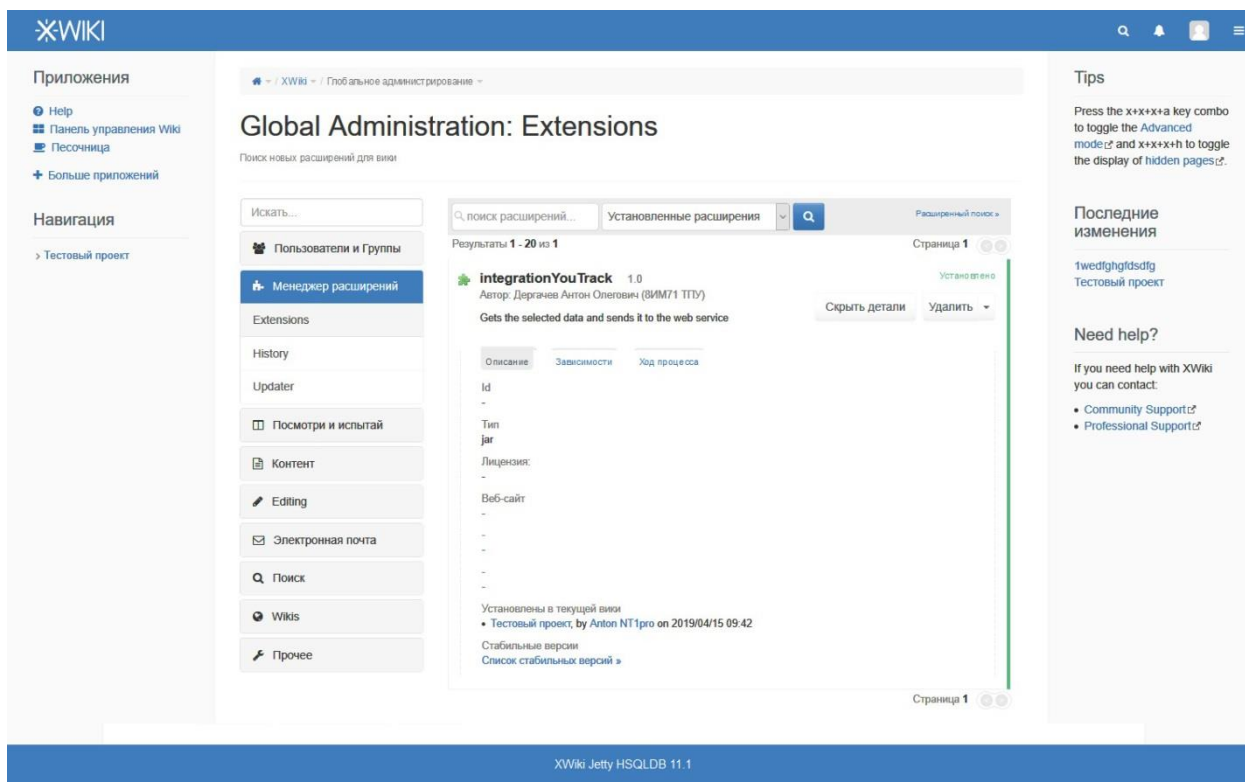


Рисунок 3.13 – Панель расширений XWiki.

Панель расширений (модулей) вики системы XWiki показывает подключенные расширения. На рисунке представлена информация о разработанном расширении (Автор расширения, название установленного проекта, его автор, тип расширения) в рамках выполнения преддипломной практики.



## **4 ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ**

### **4.1 Описание поставленных задач в разделе**

Сбор требований — один из самых важных этапов процесса создания любой информационной системы, будь то десктопное, веб или мобильное приложение, включая доработку уже существующего решения. ОАО «ТомскНИПИнефть» — ведущий корпоративный научно-проектный институт в области комплексного проектирования для нефтегазового комплекса, на предприятии ведется разработка программного обеспечения, где в ходе работы используются различные инструменты.

В частности, спецификация требований в «ТомскНИПИнефть» ведётся в XWiki, а трекинг задач осуществляется с помощью YouTrack.

Поскольку на текущий момент в «ТомскНИПИнефть» XWiki и YouTrack не имеют интеграции друг с другом, существует проблема переноса этих задач из одной системы в другую с минимальными временными затратами, и которая является актуальной на сегодняшний день, так как требует автоматизации.

Для решения данной задачи было спроектировано и разработано программное интеграционное решение, представляющее собой модуль расширения вики системы XWiki и модуль интеграции веб-сервис.

Целью данного раздела является анализ совокупности факторов, которые определяют коммерческую привлекательность разработки, ее перспективность и успешность.

Для достижения данной цели решались следующие задачи: определение потенциальных потребителей, анализ конкурентоспособности технического решения, SWOT-анализ, разработка устава проекта, планирование проекта, формирование его бюджета, а также определение финансовой и ресурсной эффективности и анализ возможных рисков.

## **4.2 Оценка коммерческого и инновационного потенциала проекта**

### **4.2.1 Потенциальные потребители продукта**

Работа в рамках магистерской выпускной квалификационной работы направлена на разработку программного обеспечения для решения интеграционной задачи передачи данных между системами XWiki и YouTrack.

Целевым рынком данной разработки являются организации, занимающиеся разработкой профильного программного обеспечения для предприятий нефтегазового комплекса.

Для того, чтобы определить, для каких организаций необходима данная разработка, нужно провести сегментирование целевого рынка.

Сегментацию можно произвести по следующим двум критериям:



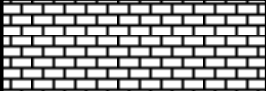
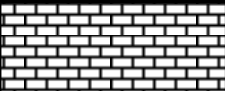
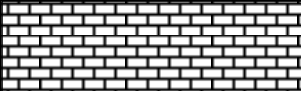
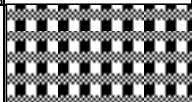

- Размер организации;
- Вариант интеграционного решения (сервиса).


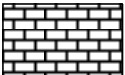

Карта сегментирования представлена в таблице 4.1.

Организациям малых размеров и молодым с экономической точки зрения не выгодно использовать модуль расширения для XWiki и веб-сервис, так как разрабатываемые и поддерживаемые ими проекты и системы уступают в сложности и масштабе проектам более крупных организаций.

Также следует учитывать тот факт, что малые компании располагаются территориально в одном помещении или рядом, в связи с этим передача данных через интернет не требуется, именно поэтому достаточно будет иметь модуль расширения для XWiki.

Таблица 4.1 - Карта сегментирования рынка услуг по разработке сервиса

		Вариант интеграционного решения			
		Настольный модуль расширения для XWiki и веб-сервис	Веб-сервис с API и встроенным модулем расширения	Модуль расширения для XWiki	Ручная передача файлов картографических данных
Размер компании	Крупные				
	Средние				
	Небольшие				

 Организация А (крупная)
  Организация Б (средняя)
  Организация В (небольшая)

Организации средних размеров могут иметь офисы, расположенные в разных регионах страны или участках планеты, поэтому передача данных через интернет необходима, для этого нужен подключенный к глобальной сети интернет веб-сервис - обмена данными. Поэтому для данных компаний подойдут первые два варианта интеграционного решения (в зависимости от уровня разработчиков и месячного оборота организации) и ручная передача данных в случае неполадок при работе программных средств.

Крупные разработчики программного обеспечения, как правило, занимаются наиболее сложными проектами, что выражается в большем объеме длительности и ресурсозатрат. Поэтому для того, чтобы разработанное программное обеспечение было качественным или же для обеспечения непрерывной работы сервиса с задействованием меньших ресурсов, необходимо проектирование интеграционного решения, включающего в себя модуль расширения XWiki и веб-сервис. Ручная передача данных также возможна, но только в случае проблем, связанных с некорректной работой программных средств.

Таким образом, разработка в рамках магистерской диссертации, заключающаяся в реализации интеграционного решения для обмена данных в задаче проектирования промышленных объектов нефтегазодобычи, подходит для организаций типа «А» и типа «Б». Компания ОАО «ТомскНИПИнефть», для нужд которой и осуществлялась разработка, относится именно к сегменту крупных компаний.

#### **4.2.2 Анализ конкурентных технических решений**

Поставленные задачи были выполнены посредством программной реализации интеграционного решения. Тем не менее, существуют готовые решения, позволяющие автоматизировать задачу передачи данных между информационными системами.

Подобные решения могут быть полезны при автоматизации простых задач, в частности задач тестирования отдельной системы на локальном уровне. Реализация поставленной задачи с помощью таких решений потребует в несколько раз больше времени и ресурсов, будет трудно-поддерживаемой с точки зрения вхождения в проект новых специалистов и реализации дополнительных тестовых сценариев в будущем.

Для подтверждения приоритетности реализации собственного решения была составлена карта сравнения конкурентных систем. В качестве основных конкурентных технических решений были выбраны следующие разработки:

- Модуль расширения для XWiki и веб-сервис (данная работа) (1);
- Программный комплекс JIRA (2);
- Сервис для визуального управления проектами Kaiten (онлайн-сервис) (3).

Результаты конкурентного анализа приведены в табл. 4.2:

Таблица 4.2 - Оценочная карта для сравнения альтернативных способов интеграции

Критерии оценки	Вес	Баллы			Конкурентоспособность		
		Б <sub>1</sub>	Б <sub>2</sub>	Б <sub>3</sub>	К <sub>1</sub>	К <sub>2</sub>	К <sub>3</sub>
1	2	3	4	5	6	7	8
<b>Технические критерии оценки ресурсоэффективности</b>							
1. Повышение производительности	0,13	5	5	3	0,65	0,52	0,39
2. Удобство в эксплуатации	0,12	4	4	1	0,48	0,48	0,12
3. Универсальность	0,11	4	5	3	0,44	0,44	0,33
4. Надежность	0,06	5	5	5	0,3	0,3	0,3
5. Потребность в ресурсах памяти	0,1	5	4	5	0,5	0,4	0,5
6. Функциональная мощность	0,03	4	5	2	0,28	0,35	0,14
7. Простота эксплуатации	0,1	4	3	2	0,4	0,3	0,2
8. Качество интеллектуального интерфейса	0,05	4	5	3	0,2	0,25	0,15
<b>Экономические критерии оценки эффективности</b>							
1. Уровень проникновения на рынок	0,05	3	3	5	0,15	0,15	0,25
2. Цена	0,2	5	3	5	0,1	0,06	0,1
3. Послепродажное обслуживание	0,03	4	3	4	0,12	0,09	0,12
4. Наличие сертификации разработки	0,02	4	4	4	0,08	0,08	0,08
Итого	1	51	49	42	3,7	3,42	2,68

Конкурентоспособность проекта 1 исполнения относительно проекта 2 исполнения:

$$K_{12} = \frac{K_1}{K_2} = \frac{3,7}{3,42} = 1,08 \quad (1)$$

Конкурентоспособность проекта 1 исполнения относительно проекта 3 исполнения:

$$K_{13} = \frac{K_1}{K_3} = \frac{3,7}{2,68} = 1,38 \quad (2)$$

Из приведённого сравнения разрабатываемого интеграционного решения с программным средством JIRA и онлайн-сервисом Kaiten можно заметить, что, несмотря на больший функционал в программном комплексе конкурента, разрабатываемый интеграционный сервис будет задействовать меньшие ресурсы памяти по сравнению с программным комплексом и меньшие функциональные (в т.ч. и человеческие) ресурсы по сравнению с

онлайн-сервисом. В связи с тем, что перечисленные ресурсы являются основополагающими, и разрабатываемое интеграционное решение будет иметь меньшую цену, то выбранный подход реализации интеграционного решения является более предпочтительным, чем использование существующих решений автоматизации передачи данных, так как конечное решение включает в себя интеграцию нескольких разнородных систем при минимизации использования ресурсов. Такое взаимодействие между разнородными системами реализуется непосредственно при использовании модуля расширения и разработанного веб-сервиса.

### 4.2.3 Технология QuaD

Технология QuaD (QUality ADvisor) – инструмент измерения характеристик, который описывает качество новой разработки, а также ее перспективность на рынке. QuaD позволяет принимать решение о целесообразности вложения капитала в НИР. Данный инструмент может использоваться при проведении различных маркетинговых исследований, существенным образом снижая их трудоемкость и повышая точность и достоверность результатов. Оценка разработки с учетом её технических и экономических особенностей, создания и коммерциализации представлена в таблице 4.3.

Таблица 4.3 - Оценочная карта QuaD

Критерии оценки	Вес	Баллы	Макс. балл	Отн. знач. (3/4)	Средне-взвеш. значение (5x2)
1	2	3	4	5	6
<b>Показатели оценки качества разработки</b>					
1. Повышение производительности труда пользователя	0,13	95	100	0,95	0,1235
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,12	80	100	0,8	0,096
3. Универсальность (форматы данных, атрибуты и геометрии)	0,11	75	100	0,75	0,0825
4. Надежность	0,06	90	100	0,9	0,0054

### Продолжение таблицы 4.3

5. Потребность в ресурсах памяти	0,1	95	100	0,95	0,095
6. Функциональная мощность (предоставляемые возможности)	0,07	75	100	0,75	0,0525
7. Простота эксплуатации	0,1	85	100	0,85	0,085
8. Качество интеллектуального интерфейса	0,05	70	100	0,7	0,035
<b>Показатели оценки коммерческого потенциала разработки</b>					
1. Конкурентоспособность продукта	0,05	80	100	0,8	0,04
2. Уровень проникновения на рынок	0,05	65	100	0,65	0,0325
3. Цена	0,2	100	100	1	0,2
4. Послепродажное обслуживание	0,03	80	100	0,8	0,024
5. Срок выхода на рынок	0,01	60	100	0,6	0,006
6. Наличие сертификации разработки	0,02	75	100	0,75	0,015
Итого					0,8924

Оценка качества и перспективности по технологии QuaD определяется по формуле:

$$P_{cp} = \sum V_i B_i \quad (3)$$

где  $P_{cp}$  – средневзвешенное значение показателя качества и перспективности научной разработки;  $V_i$  – вес показателя (в долях единицы);  $B_i$  – балл  $i$ -го показателя.

В результате проведенного анализа выяснено, что перспективность разработки равна 89,24%. Поскольку значение показателя входит в промежуток от 80% до 100%, то данная разработка является перспективной.

#### 4.2.4 SWOT-анализ

Описание сильных и слабых сторон проекта, выявление возможностей и угроз. Это позволит использовать сильные стороны проекта, чтобы максимизировать возможности и свести к минимуму угрозы и слабости при разработке. Результаты первого этапа анализа представлены в табл. 4.4.

Таблица 4.4 - Результаты первого этапа SWOT-анализа

	<p>Сильные стороны</p> <p>S1. Наличие открытой документации для создания модуля расширения к XWiki.</p> <p>S2. Возможность работы отдела аналитиков и информационных технологий удаленно территориально.</p> <p>S3. Расположение модуля расширения в XWiki для удобной работы проектировщика.</p> <p>S4. Высокая скорость работы.</p>	<p>Слабые стороны</p> <p>W1. Не возможность передачи заголовков кириллицы.</p> <p>W2. Происходит перекодирование данных при отправке из разных кодировок в utf-8.</p> <p>W3. Работа модуля расширения в XWiki усложняет разработку интеграции из-за проблем с передачей данных.</p>
<p>Возможности</p> <p>O1. Появление дополнительного спроса на новый продукт.</p> <p>O2. Повышение стоимости конкурентных разработок.</p>	<p>Направления развития</p> <p>O1S1. Возможность рассылки и выкладывания в общий доступ документации к разработке проекта.</p> <p>O2S2S3S4. Появление новых оптимизаций, которые в будущем можно будет легко интегрировать в разработку, тем самым значительно улучшая ее конкурентоспособность.</p>	<p>Сдерживающие факторы</p> <p>O1W1. Несмотря на не возможность передачи заголовков кириллицей, существуют алгоритмы перевода кириллицы в латиницу.</p> <p>O2W1. Перекодирование происходит на программном уровне, используя стандартные средства предоставляемых языком программирования.</p>
<p>Угрозы</p> <p>T1. Отсутствие финансирования и привлеченной рабочей силы ограничит функционал проекта</p> <p>T2. Нехватка опытных разработчиков в команде по сравнению с конкурентными разработками приведет к ошибкам на стадиях проектирования и реализации.</p>	<p>Угрозы развития</p> <p>T1S1. Документация не всегда описывается достаточно понятно для разработчика, требуется большого опыта разработки.</p> <p>T2S2S3. Широкие возможности для масштабирования проекта доступны опытным разработчикам, возникновения спорных ситуаций при масштабировании проекта разными отделами предприятия.</p>	<p>Уязвимости</p> <p>T1W1. Огромная трудоемкость работы и большое число разнообразных ошибок при разработке. При прекращении работы проект не сможет раскрыть своего потенциала и повысить эффективность, в связи с чем варианты конкурентов могут пользоваться большим спросом.</p> <p>T2W3. Использование привычной методологии проектирования потребует большой работы для увеличения масштабности проекта, что увеличит его стоимость.</p>



Таким образом, можно сделать вывод, что основными рисками проекта является недостаточное количество ресурсов и наличие конкурентов со сходным и превосходящим функционалом, но при этом сильными факторами являются удобство работы с программой и уменьшение трудовых ресурсов при использовании данной системы.

#### 4.2.5 Диаграмма Исикавы

Одной из проблем предприятия, для решения которой выполнялась данная работа, является передача (в т.ч. и обмен) данных между разнородными системами XWiki и YouTrack. Для описания причин проблемы была использована диаграмма Исикавы.

Диаграмма Исикавы — графический способ исследования и определения наиболее существенных причинно-следственных взаимосвязей между факторами и последствиями в исследуемой ситуации или проблеме. Диаграмма представлена на рисунке 4.1.

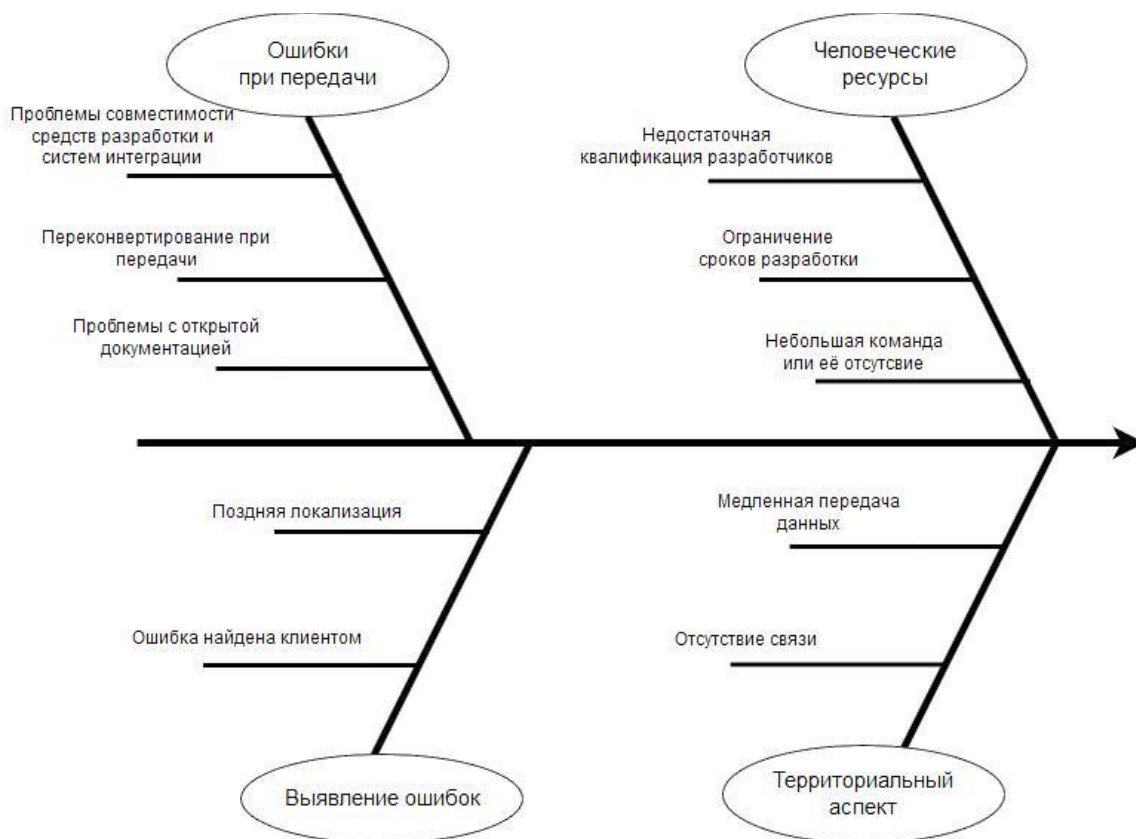


Рисунок 4.1 - Диаграмма Исикавы

По построенной диаграмме Исикавы можно сделать следующие выводы о наиболее влияющих факторах: для устранения проблемы неудобства пользователей в фиксировании задач по разработке ПО в первую очередь необходимо мотивировать разработчиков на решении известных проблем; интегрировать системы с учетом пожеланий пользователей.

### 4.3 Инициация проекта

В рамках процессов инициации определяются цели и содержание и фиксируются изначальные финансовые ресурсы. Определяются внутренние и внешние заинтересованные стороны проекта, которые будут взаимодействовать и влиять на общий результат научного проекта.

#### 4.3.1 Цели и результат проекта

Перед определением целей необходимо перечислить заинтересованные стороны проекта. Информация по заинтересованным сторонам представлена в таблице 4.5.

Таблица 4.5 - Заинтересованные стороны проекта

<b>Заинтересованные стороны проекта</b>	<b>Ожидание заинтересованных сторон</b>
Организация-заказчик	Автоматизация процесса обмена данными между XWiki и YouTrack в задачах планирования и управления проектами разработки ПО
Пользователь (проектировщик)	Возможность создавать задачи в YouTrack из описания их в XWiki
Студент Научный руководитель	Готовая магистерская диссертация

Цели и результат проекта в части выполнения одной из двух основных задач магистерской диссертации, а именно реализации интеграционного решения для автоматизации процесса обмена данными в задачах проектирования и управления проектами разработки ПО, представлены в таблице 4.6.

Таблица 4.6 - Паспорт проекта

Название проекта	Разработка модуля интеграции корпоративной базы знаний на платформе XWiki и системы трекинга задач YouTrack.
Суть проекта	Поиск интеграционного решения для обмена данными между разнородными системами XWiki и YouTrack для повышение эффективности управления процессом разработки ПО.
Бизнес-окружение проекта	Проектные институты нефти и газа, организации по разработке IT-решений в области управления процессом разработки ПО
Цели проекта	Постановка задачи и описание текущих бизнес-процессов предприятия Оптимизация бизнес-процессов планирования и разработки ПО. Поиск интеграционного решения для обмена данными между системами XWiki и YouTrack Реализация интеграционного решения для обмена данными между системами XWiki и YouTrack
Ожидаемый результат проекта	Отображение описания задач из XWiki в системе трекинга задач YouTrack
Критерии приемки результата проекта	Корректная передача данных выбранного пользователем требования, успешное создание задачи в системе трекинга задач YouTrack.
Требования к результату проекта	Модуль расширения работает согласно плану Пользователь выделяет описание требования в системе XWiki, создаёт страницу требования с выделенным контентом. Происходит отправка данных в формате json на веб сервис (заголовок, контент, автор, url, ключ), веб сервис используя REST API создаёт задачу в системе YouTrack согласно принятым данным.
Бюджет проекта	350000 рублей
Сроки проекта	28.01.2019 – 07.06.2019

### 4.3.2 Организационная структура проекта

Организационная структура проекта представлена в таблице 4.7.

Таблица 4.7 - Организационная структура проекта

№	ФИО, основное место работы, должность	Роль в проекте	Функции
1	Дергачев Антон Олегович, студент, соискатель на должность программиста ОАО «ТомскНИПИнефть»	Системный аналитик, разработчик	<ul style="list-style-type: none"><li>• Изучение предметной области</li><li>• Поиск решения для выполнения поставленной задачи</li><li>• Проектирование архитектуры</li><li>• Реализация</li><li>• Тестирование</li></ul>
2	Копнов Максим Валерьевич, ОАО «ТомскНИПИнефть», заведующий сектором разработки ПО	Заказчик, консультант	<ul style="list-style-type: none"><li>• Постановка задачи</li><li>• Интервью по работе подразделения в процессе по разработки ПО</li><li>• Проверка разработки</li></ul>
3	Мирошниченко Евгений Александрович, к.т.н., доцент отделения ИТ	Научный руководитель	<ul style="list-style-type: none"><li>• Помощь в поиске интеграционного решения</li><li>• Составление научных целей и задач</li><li>• Проверка документации</li></ul>

В данном разделе были выявлены заинтересованные стороны проекта, составлен паспорт проекта, и сформулирована организационная структура проекта. Так же указаны сроки выполнения проекта с 28.01.2019 по 07.06.2019, сформирован бюджет проекта 350000 тыс. руб., распределены роли в организационной структуре и описаны их функции.

## 4.4 Планирование научно-исследовательских работ

### 4.4.1 Структура работ в рамках научно-технического

#### исследования

Перечень этапов и работ в рамках проведения научно-технического исследования представлен в табл. 4.8:

Таблица 4.8. - Перечень этапов, работ и распределение исполнителей

Основные этапы	№	Содержание работ	Исполнитель
Постановка задачи	1	Формирование проектного задания	Р, И <sup>1</sup>
	2	Выявление проблемы, возникающей при решении задачи	Р

<sup>1</sup> Р – Научный руководитель; И – Исполнитель (системный аналитик и программист).

Продолжение таблицы 4.8

Системный анализ предприятия	3	Консультирование по работе структурных подразделений предприятия	Р, И
	4	Описание бизнес-процессов предприятия при решении поставленной задачи	И
	5	Оптимизация бизнес-процессов предприятия при решении поставленной задачи	И
Разработка технического задания	6	Сбор информации и пожеланий от заказчика	Р, И
	7	Анализ требований	И
	8	Документирование требований	И
	9	Проверка требований	И
Аналитический обзор	10	Подбор и изучение материалов по обмену данными	И
	11	Изучение уже существующих интеграционных решений в данной области	И
	12	Выбор инструментов и средств для реализации интеграционного решения	И
Проектирование интеграционного решения	13	Проектирование функционала системы	И
	14	Проектирование макетов и сценария использования интеграционного решения	И
Реализация интеграционного решения	15	Выбор языка программирования и инструментов разработки для интеграционного решения	И
	16	Реализация модуля расширения XWiki	И
Обобщение и оценка результатов	17	Тестирование работы модуля расширения и (в т.ч. исправление ошибок)	И
	18	Проверка работоспособности системы на нескольких экспериментах	И
	19	Обобщение результатов работы и составление руководства пользователя	Р, И

#### 4.4.2 Определение трудоемкости и графика выполнения работ

Оценим трудоемкость выполнения вышеозначенных работ. Для этого оценим минимальное и максимальное время выполнения каждой работы.

Также произведем расчет ожидаемого значения трудоемкости по следующей формуле:

$$t_{ож,i} = \frac{(3t_{min,i} + 2t_{max,i})}{5} \quad (4)$$

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

$t_{\min, i}$  – минимально возможная трудоемкость выполнения  $i$ -ой работы (оптимистическая оценка: в предположении наиболее благоприятного стечения обстоятельств), чел.-дн.;

$t_{\max, i}$  – максимально возможная трудоемкость  $i$ -ой работы (пессимистическая оценка: в предположении наиболее неблагоприятного стечения обстоятельств), чел.-дн.

Следует заметить, что исполнитель задействован в каждой из перечисленных работ, а потому невозможно ускорение за счет параллельности их выполнения.

На основе таб. 4.7 строится календарный план-график научного исследовательского проекта. График строится для максимального по длительности исполнения работ в рамках научно-исследовательского проекта с разбивкой по месяцам и неделям за период выполнения данной диссертационной работы. Для удобной визуализации разработка календарного плана графика производится в бесплатной программе GanttProject в виде диаграммы Ганта (рис. 4.1).

Таблица 4.9 - Временные показатели научно-технического исследования

№	Содержание работ	Мин. время выполнения (дн.)			Макс. время выполнения (дн.)			$t_{ож}$	Занятость ресурсов (проценты)	Длительность работ в рабочих днях			Длительность в календарных днях		
		И1	И2	И3	И1	И2	И3			И	Р	З	И	Р	З
1	Формирование проектного задания	0,5			1			0,7	Заказчик (50%), исполнитель (50%)	1	0	1	1	0	1
2	Выявление проблемы, возникающей при решении задачи	0,5			1			0,7	Руководитель (40%), исполнитель (60%)	1	1	0	1	1	0
3	Консультирование по работе структурных подразделений предприятия	3			6			4,2	Заказчик (50%), исполнитель (50%)	5	0	5	6	0	6
4	Описание бизнес-процессов предприятия при решении поставленной задачи	5			7			5,8	Исполнитель (100%)	7	0	0	8	0	0
5	Оптимизация бизнес-процессов предприятия при решении поставленной задачи	15			30			21	Исполнитель (100%)	25	0	0	31	0	
6	Сбор информации и пожеланий от заказчика	1			3			1,8	Заказчик (50%), исполнитель (50%)	2	0	2	3	0	3
7	Анализ требований	4			7			5,2	Исполнитель (100%)	6	0	0	8	0	0
8	Документирование требований	5			8			6,2	Исполнитель (100%)	7	0	0	9	0	0
9	Проверка требований	5			8			6,2	Исполнитель (100%)	7	0		9	0	0
10	Подбор и изучение материалов по обмену данных между системами	1			4			2,2	Исполнитель (100%)	3	0	0	4	0	0
11	Изучение уже существующих интеграционных решений в данной области	2			3			2,4	Исполнитель (100%)	3	0	0	4	0	0

Продолжение таблицы 4.9

12	Выбор инструментов и средств для реализации системы	0,5	1	0,7	Исполнитель (100%)	1	0	0	1	0	0	
13	Изучение технической документации по форматам данных	2	4	2,8	Исполнитель (100%)	3	0	0	4	0	0	
14	Проектирование функционала системы	1	2	1,4	Исполнитель (100%)	2	0	0	4	0	0	
15	Проектирование макетов и сценария использования интеграционного решения	2	4	2,8	Исполнитель (100%)	3	0	0	1	0	0	
16	Выбор языка программирования и среды разработки	1	1	1	Исполнитель (100%)	1	0	0	1	0	0	
17	Реализация модуля расширения Xwiki	10	15	12	Исполнитель (100%)	14	0	0	22	0	0	
18	Тестирование работы модуля расширения	2	4	2,8	Исполнитель (100%)	3	0	0	6	0	0	
19	Проверка работоспособности системы	1	1	1	Исполнитель (100%)	1	0	0	1	0	0	
20	Обобщение результатов работы и составление руководства пользователя	1	2	1,4	Руководитель (25%), исполнитель (75%)	2	0	0	3	1	0	
									Итого:	127	2	10



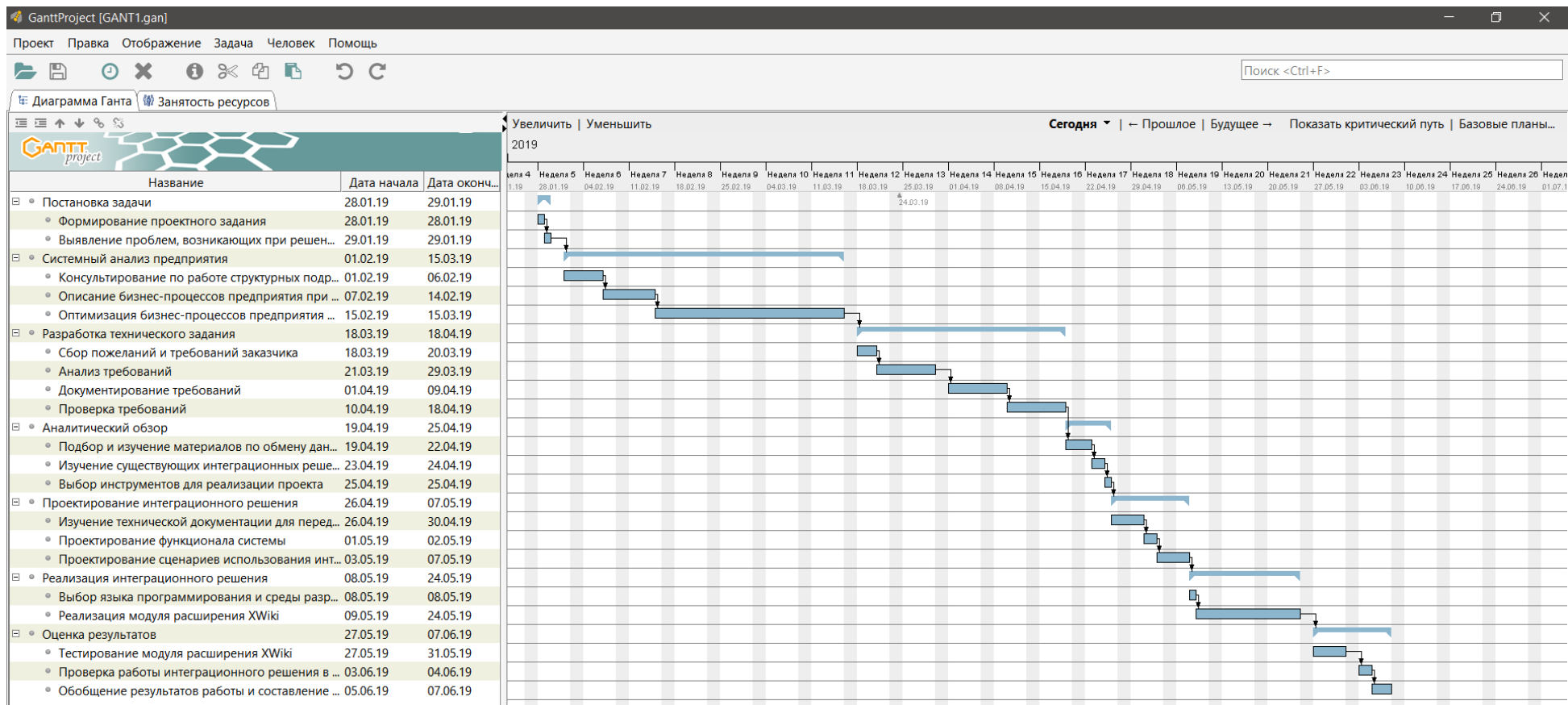


Рисунок 4.2 - Календарный план для выполнения научно-исследовательского проекта

### **4.4.3 Бюджет научно-технического исследования**

В состав бюджета входит стоимость всех расходов, необходимых для выполнения работ по магистерской диссертации. При формировании бюджета используется группировка затрат по следующим статьям:

- Материальные затраты;
- Основная заработная плата исполнителей темы;
- Дополнительная заработная плата исполнителей темы;
- Отчисления во внебюджетные фонды (страховые отчисления);
- Накладные расходы.

#### **4.4.3.1 Расчёт материальных затрат**

Данная статья расходов включает стоимость всех материалов, используемых при разработке в рамках выполнения данной выпускной квалификационной работы. При выполнении работы был использован один персональный компьютер в компании. Мелкие расходы (канцелярия, затраты на печать и пр.) могут быть отнесены к статье прочих расходов.

#### **4.4.3.2 Амортизационные отчисления**

Поскольку для проведения исследований специальное дорогостоящее оборудование не приобреталось, при расчете затрат учитывается только амортизация. Первоначальная стоимость ПК магистранта, используемого для проведения исследований, составляет 50000 рублей. Срок полезного использования данной машины – 3 года, из которых 4 месяцев машина использовалась для написания ВКР.

Норма амортизации:

$$A_n = \frac{1}{n} * 100\% = \frac{1}{3} \times 100\% = 33,33\% \quad (2)$$

Годовые амортизационные отчисления:

$$A_g = 50000 \times 0,33 = 16500 \text{ рублей} \quad (3)$$

Ежемесячные амортизационные отчисления:

$$A_m = \frac{16500}{12} = 1375 \text{ рублей} \quad (4)$$

Итоговая сумма амортизации основных средств:

$$A = 1375 \times 4 = 5500 \text{ рублей} \quad (5)$$

Таким образом затраты на амортизацию ПК составляют 5500 рублей.

#### 4.4.3.3 Расчёт основной заработной платы исполнителей

В данную статью расходов включается основная заработная плата научного руководителя, руководителя от предприятия и студента. Расчёт выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя. Трудоёмкость всех исполнителей в разные промежутки времени на протяжении выполнения магистерской диссертации была просуммирована и представлена в виде затраченных дней. Таким образом, если согласно плану, работа над проектом должна вестись в течение 123 дней, то реально затраченное время каждого исполнителя в днях отличается от данной цифры, а также от того, что можно увидеть из диаграммы Ганта (на ней декомпозиция происходит с точностью до дней).

Расчет основной заработной платы приведен в таблице 4.10.

Таблица 4.10 - Основная заработная плата исполнителей проекта

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб.	Затраты времени, раб.дни	Районный коэф.	Основная заработная плата, руб.
Научный руководитель	29568	1374,5	2	1,3	8934,25
Инженер	21760	873,2	127		144165,32
Заказчик	50000	2511,2	10		32645,6
Итого по статье С <sub>осн</sub>					185745,17

#### 4.4.3.4 Расчёт дополнительной заработной платы исполнителей

В данную статью включается сумма выплат, предусмотренных законодательством о труде. Например, оплата очередных и дополнительных отпусков; оплата времени, связанного с выполнением государственных и общественных обязанностей; выплата вознаграждения за выслугу лет и т. п. (в среднем — 12% от суммы основной заработной платы). Расчёты дополнительной заработной платы приведены в таблице 4.11.

Таблица 4.11 - Дополнительная заработная плата исполнителей проекта

Исполнитель	Основная заработная плата, руб.	Районный коэф.	Дополнительная заработная плата, руб.
Научный руководитель	8934,25	0,12	1072,11
Инженер	144165,32		17299,83
Заказчик	32645,6		3917,47
Итого по статье С <sub>доп</sub>			22289,42

#### 4.4.3.5 Расчёт отчислений во внебюджетные фонды

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.

На 2019 г. в соответствии со ст. 425, 426 НК РФ действуют следующие тарифы страховых взносов: ПФР — 0.22 (22%), ФСС РФ — 0.029 (2,9%), ФФОМС — 0,051 (5,1%). Рассчитанные отчисления представлены в таблице 4.12.

Таблица 4.12 - Отчисления во внебюджетные фонды

Исполнитель	Заработная плата, руб.	Отчисления, руб.			Сумма отчислений, руб.
		ПФР (22%)	ФСС (2,9%)	ФФОМС (5,1%)	
Научный руководитель	4002,5	880,6	116,1	204,1	1200,8
Инженер	55121,4	12126,7	1598,5	2811,2	16536,4
Заказчик	14625,1	3217,5	424,1	745,9	4387,5
Итого по статье С <sub>внеб</sub>					22124,7

#### 4.4.3.6 Расчёт накладных расходов

Накладные расходы учитывают все затраты, не вошедшие в предыдущие статьи расходов: печать и ксерокопирование, оплата электроэнергии, оплата пользования услугами интернета.

Перечисленные расходы требуют низких затрат денежных средств относительно заработной платы исполнителей, поэтому величина коэффициента накладных расходов  $k_{\text{накл}}$  была принята в размере 5%.

Расчёт накладных расходов ведётся по формуле:

$$C_{\text{накл}} = k_{\text{накл}} * C_{\text{зп}} = 0,16 * 73752 = 11800,32 \quad (5)$$

#### 4.4.3.7 Формирование бюджета проекта

Согласно произведённым расчётам, сумма затрат по всем статьям расходов была рассчитана и представлена в качестве общего бюджета проекта в таблице 4.13.

Таблица 4.13 - Бюджет проекта

Статья затрат	Сумма, руб.
Амортизационные отчисления	185745,17
Заработная плата исполнителей	22124,7
Отчисления во внебюджетные фонды	11800,32
Накладные расходы	30912,86
Планируемая прибыль	50116,61
НДС	185745,17
Итого	300699,66

Рассчитанный бюджет не превышает бюджета в 130000 рублей, указанного в выдержках из устава проекта (табл. 4.6).

Планируемая прибыль: 50116,61 руб.

НДС: 185745,17 руб.

Итоговая цена: 300699,66 руб.

#### 4.4.3.8 Риски проекта

Риски проекта - это возможные непредвиденные события, которые могут возникнуть в проекте и вызвать негативные последствия, которые повлекут за собой отрицательные эффекты на весь проект.

Риски, которые могут возникнуть при реализации проекта, приведены в таблице 4.14

Таблица 4.14 - Риски проекта

Риск	Потенциальное Воздействие	Вероятность Наступления	Влияние риска	Уровень риска	Условия наступления	Мероприятия по устранению
Отсутствие финансирования	Снижение качества продукта и увеличение сроков выхода на рынок	Средняя	Среднее	Высокий	Недостаточная заинтересованность заказчика реализации проекта, недостаточное привлечение капитала	Поиск возможных инвесторов
Недостаток кадров	Увеличение сроков выхода на рынок, снижение качества продукта	Низкая	Высокое	Средней	Отсутствие специалистов в регионе, недостаток финансирования	Увеличение команды разработчиков
Низкая масштабируемость разработки	Снижение качества продукта	Низкая	Низкое	Среднее	Недостаток высококвалифицированных кадров в команде разработчиков, отсутствие финансирования	Привлечение высококвалифицированных специалистов в команду разработчиков
Ошибки при проектировании взаимодействия системы	Снижение качества продукта	Средняя	Среднее	Высокое	Недостаток высококвалифицированных кадров в команде разработчиков, отсутствие финансирования	Увеличение команды проектировщиков

#### 4.5 Определение финансовой и ресурсной эффективности проекта

Определение эффективности производится путем определения интегрального показателя эффективности научного исследования через нахождение величин финансовой и ресурсной эффективности. Интегральный финансовый показатель определяется по следующей формуле:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{p,i}}{\Phi_{\text{max}}} \quad (6)$$

где  $I_{\text{финр}}^{\text{исп.}i}$  — интегральный финансовый показатель разработки;  $\Phi_{p,i}$  — стоимость  $i$ -го варианта исполнения;  $\Phi_{\text{max}}$  — максимальная стоимость исполнения научно-исследовательского проекта.

Таким образом, интегральный финансовый показатель данной разработки, выполненной по варианту исполнения  $\frac{\Phi_{p,i}}{\Phi_{\text{max}}} = \frac{300699,66 \text{ руб.}}{350000 \text{ руб.}} = 0,91$ .

Интегральный показатель ресурсоэффективности можно определить следующим образом:

$$I_{p,i} = \sum a_i b_i \quad (7)$$

где  $I_{p,i}$  — интегральный показатель ресурсоэффективности для  $i$ -го варианта разработки,  $a_i$  — весовой коэффициент  $i$ -го варианта разработки,  $b_i$  — бальная оценка  $i$ -го варианта исполнения разработки, устанавливаемая экспертным путем по выбранной шкале оценивания,  $n$  — число параметров сравнения.

В связи с представленными исполнениями проекта был произведен перерасчет весовых коэффициентов технических критериев для оценки ресурсо-эффективности проекта.

Таблица 4.15 - Сравнительная оценка характеристик вариантов исполнения проекта

№	Критерии	Весовой коэф.	Баллы			Интегральные показатели ( $I_{p,i}$ )		
			$I_1$	$I_2$	$I_3$	$I_1$	$I_2$	$I_3$
1	Повышение производительности труда пользователя	0,15	5	5	5	0,75	0,75	0,75
2	Удобство в эксплуатации	0,15	5	5	5	0,75	0,75	0,75
3	Универсальность	0,13	5	5	5	0,65	0,65	0,65
4	Надежность	0,12	5	4	5	0,6	0,48	0,6
5	Потребность в ресурсах памяти	0,11	5	4	5	0,55	0,44	0,55
6	Функциональная мощность (предоставляемые возможности)	0,14	5	5	5	0,7	0,7	0,7
7	Простота эксплуатации	0,1	5	5	5	0,5	0,5	0,5
8	Качество интеллектуального интерфейса	0,1	5	4	4	0,5	0,4	0,4
Ресурсоэффективность		1				5	4,67	4,9

Согласно расчётам, приведённым в таблице 4.15, можно сделать вывод, что самым эффективным исполнением с точки зрения ресурсоэффективности является первое (фактическое) исполнение  $I_1$  (с показателем конкурентоспособности  $I_{p,i} = 5$ ), по которому и был реализован проект в рамках магистерской диссертации.

#### 4.6 Вывод по разделу

В ходе выполнения данной работы было выявлено, что исследование, проведенное в рамках выполнения выпускной квалификационной работы, относится к сегменту разработки программного обеспечения, потребителем которого являются коммерческие организации, создающие различное программное обеспечение и использующие в процессе разработки и управления проектами ПО такую систему как XWiki

Конкурентными преимуществами работы могут стать меньшая потребность в ресурсах памяти и простота в эксплуатации. Оценка готовности проекта к коммерциализации показала, что перспективность разработки данного проекта выше среднего, а следовательно исследование необходимо продолжать.



SWOT-анализ показал, что ключевой проблемой проекта является отсутствие финансирования, из-за чего затруднительно активно внедрять и расширять данную систему. Рассмотрев слабые стороны проекта и его возможности, можно прийти к тому же выводу, что необходима стратегия поиска дополнительного финансирования проекта, которое позволит улучшать технологии разработанной системы, что положительно скажется на автоматизации бизнес-процессов предприятия, где система будет интегрирована.

В ходе расчета сметы затрат на выполнение проекта была выявлена цена разработки проекта, которая равна 300699 руб.

Вероятность наступления рисков ниже среднего, однако, данные риски имеют высокое влияние на проект.

Таким образом, было выявлено, что разработка данного проекта является достаточно перспективной.

## 5 СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

### 5.1 Правовые и организационные вопросы обеспечения безопасности

Правовую основу охраны труда составляют законодательные акты и нормативные правовые документы по охране труда, имеющие иерархическую структуру, обладающие различной юридической силой. Нормы трудового права регулируют любые отношения, связанные с использованием личного труда.

Основной задачей трудового договора является создание необходимых организационно-правовых условий для достижения оптимального согласования интересов сторон трудовых отношений.

По заключенному, на основании Трудового кодекса РФ от 30.12.2001 № 197-ФЗ, трудовому договору работник обязан:

- Лично выполнять определенную трудовым договором функцию с соблюдением внутреннего трудового распорядка, установленного Работодателем;
  - Добросовестно исполнять свои трудовые обязанности, приказы, распоряжения и указания своего непосредственного руководителя;
  - Бережно относиться к имуществу Работодателя, в том числе находящемуся в его пользовании, правильно и по назначению использовать переданные ему для работы оборудование, приборы;
  - Нести иные обязанности, предусмотренные действующим законодательством Российской Федерации и локальными нормативными актами Работодателя.
- В свою очередь работодатель обязан:
  - Предоставить работнику работу по обусловленной трудовой функции,
  - Обеспечить условия труда, предусмотренные трудовым законодательством и иными правовыми нормативными актами,

соглашениями, содержащими нормы трудового права, коллективным договором;

- Нести иные обязанности, предусмотренные действующим законодательством Российской Федерации и локальными нормативными актами Работодателя.

- Также, работник имеет право:

- Изменять и расторгать трудовой договор в порядке и на условиях, определенных трудовым законодательством Российской Федерации,

- Своевременно и в полном объеме получать заработную плату в соответствии со своей квалификацией, сложностью труда, количеством и качеством выполненной работы,

- Работодатель имеет право:

- Изменять и расторгать договор в порядке и на условиях, определенных трудовым законодательством Российской Федерации,

- Требовать от Работника и контролировать исполнение им трудовых обязанностей и бережного отношения к имуществу нормативных актов, трудовой дисциплины, правил техники безопасности, производственной санитарии и противопожарной защиты и иных локальных нормативных актов Общества;

- Привлекать Работника к дисциплинарной и материальной ответственности за неисполнение или ненадлежащее исполнение Работником трудовых обязанностей в порядке, установленном трудовым законодательством Российской Федерации.

На основании ТК РФ статьи 135, трудовым договором устанавливается заработная плата в соответствии с действующими у данного работодателя системами оплаты труда. Заработная плата выплачивается Работнику два раза в месяц в порядке и сроки, установленные Правилами внутреннего трудового распорядка в соответствии с действующим

законодательством: за первую половину месяца – 30 числа текущего месяца, окончательный расчет – 15 числа месяца следующего за отчетным.

В соответствии с ГОСТ 12.2.032-78 «Система стандартов безопасности труда (ССБТ)» к рабочему месту предъявляются следующие основные эргономические требования:

– Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы;

- Конструкцией рабочего места должно быть обеспечено выполнение трудовых операций в пределах зоны досягаемости моторного поля;

- Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы;

- Рабочее место для выполнения работ сидя организуют при легкой работе, не требующей свободного передвижения работающего, а также при работе средней тяжести в случаях, обусловленных особенностями технологического процесса;

- При проектировании оборудования и организации рабочего места следует учитывать антропометрические показатели женщин (если работают только женщины) и мужчин (если работают только мужчины); если оборудование обслуживают женщины и мужчины - общие средние показатели женщин и мужчин;

- Конструкцией рабочего места должно быть обеспечено оптимальное положение работающего, которое достигается регулированием высоты рабочей поверхности, сиденья и пространства для ног.

## 5.2 Производственная безопасность

Выявление на рабочем месте факторов производственной среды и трудового процесса, источников вредных и (или) опасных факторов осуществляется путем изучения представляемых работодателем:

- Технической (эксплуатационной) документации на производственное оборудование (машины, механизмы, инструменты и приспособления), используемое работником на рабочем месте;
- Должностной инструкции и иных документов, регламентирующих обязанности работника;
- Характеристик применяемых в производстве материалов и сырья;
- Результатов ранее проводившихся на данном рабочем месте исследований (испытаний) и измерений вредных и (или) опасных факторов.

Основные вредные и опасные факторы при выполнении технологических процессов разработки программного обеспечения представлены в таблице 5.1.

Таблица 5.1 - Возможные опасные и вредные факторы

Факторы (ГОСТ 12.0.003-2015)	Этапы работ			Нормативные документы
	Разраб отка	Изгото вление	Эксплу атация	
1. Отклонение показателей микроклимата	-	-	-	СанПиН 2.2.4.548–96. Гигиенические требования к микроклимату производственных помещений.
2. Превышение уровня шума	-	-	-	ГОСТ 12.1.029-80 Средства и методы защиты от шума.
3. Отсутствие или недостаток естественного света	+	-	+	СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95
4. Повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека	-	-	-	ГОСТ 12.1.038–82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов

## 5.3 Анализ опасных и вредных производственных факторов

### 5.3.1 Отклонение показателей микроклимата

Микроклимат помещения – это комплекс метеорологических условий в данном помещении. Компьютеры могут привести к увеличению температуры и снижению относительной влажности в помещении. В СанПиН 2.2.4.548 – 96 установлены величины параметров микроклимата, создающие комфортные условия [19].

Требования к микроклимату определяются исходя из категории тяжести работ. Работа разработчика-программиста относится к легкой категории 1Б [19], оптимальные и допустимые значения микроклимата представлены в таблице 5.2.

Таблица 5.2 - Оптимальные и допустимые значения микроклимата

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	23-25	20-24	40-60	0,1
Тёплый	20-22	21-25	40	0,1

Если температура воздуха отличается от нормальной, то время пребывания в таком помещении должно быть ограничено в зависимости от категории тяжести работ. Температура в рассматриваемом помещении в холодное время года может опускаться до 19-21 °С, а в теплое время года подниматься до 25-28 °С. Данные показатели соответствуют допустимым значениям температуры, рекомендуемое время работы при температуре воздуха ниже допустимых величин представлено в таблице 5.3, при температуре выше допустимых величин представлено в таблице 5.4.

Таблица 5.3 – Рекомендуемое время работы при температуре воздуха ниже допустимых величин

Температура воздуха, °С	Время пребывания, не более, ч
17,0	6
18,0	7

Таблица 5.4 – Рекомендуемое время работы при температуре воздуха выше допустимых величин

Температура воздуха, °С	Время пребывания, не более, ч
30,0	5
29,5	5,5
29,0	6

К мероприятиям по оздоровлению воздушной среды в производственном помещении относятся правильная организация вентиляции и кондиционирования воздуха, отопление помещений. В рассматриваемом помещении вентиляция осуществляется естественным и механическим путём. В зимнее время в помещении предусматривается система отопления. Это обеспечивает нормальное состояние здоровья работников в помещении.

### 5.3.2 Превышение уровня шума

Для исследуемого объекта (модуля интеграции вики системы XWiki) основными источниками шумов, а также и вибрации являются производственные процессы, сопровождающиеся шумом, такая как, работа с компьютером. Говоря, о действии шума на организм, следует иметь в виду, что он оказывает как местное, так и общее воздействие. Неблагоприятно отражается шум на нервной системе, вызывая головные боли, бессонницу, ослабление внимания, замедление психических реакций, пониженный

аппетит, боли в ушах, что, в конечном счете, приводит к понижению работоспособности [20].

Допустимые уровни звука и звукового давления для рабочего места разработчика-программиста согласно ГОСТ 12.1.003-83 ССБТ «Общие требования безопасности» представлены в таблице 5.5 [21].

Таблица 5.5 – Предельно допустимые уровни звука

Вид трудовой деятельности/ Частоты	Уровни звука и звукового давления, дБ, в октавных полосах со среднегеометрическими частотами, Гц								
	31,5	63	125	250	500	1000	2000	4000	8000
Научная деятельность, проектирование, программирование, Рабочие места проектно-конструкторских бюро, программистов вычислительных машин и т.д.	86	71	61	54	49	45	42	40	38

Допустимый уровень звукового давления колеблется от 38 дБ до 86 дБ при частоте от 8000 Гц до 31,5 Гц, соответственно.

Для снижения уровня шума, производимого персональными компьютерами, рекомендуется регулярно проводить их техническое обслуживание: чистка от пыли, замена смазывающих веществ; также применяются звукопоглощающие материалы [22].



### 5.4.3 Отсутствие или недостаток естественного света

Оценка освещенности рабочей зоны необходима для обеспечения нормативных условий работы в помещениях и проводится в соответствии с СанПиН 2.2.1/2.1.1.1278-03.

Освещение в недостаточной степени может привести к напряжению зрения, ослаблению внимания и наступлению преждевременной утомленности. Слепление, резь в глазах и раздражение могут быть вызваны чрезмерно ярким освещением. Свет на месте труда может создать сильные тени или отблески, а также дезориентировать работающего. Основным документом, регламентирующим нормы освещенности, является СНиП 23-05-95\* [23].

В производственном помещении используется искусственное и естественное освещение. Естественное освещение – боковое, осуществляемое через оконные проемы в боковых наружных стенах. Искусственное освещение – общее равномерное (световой поток равномерно распределяется по помещению без учета расположения оборудования),

Основным показателем качества освещения является освещенность  $E$  - поверхностная плотность светового потока. По характеристике зрительной работы труд программиста относится к разряду III подразряду Г (высокой точности), т.е. наименьший размер объекта различения от 0,3 до 0,5 мм (точка) [24]. Это значит, что нормативное значение освещенности рабочего места должно быть 200 лк [24].

Для обеспечения нормативной освещенности необходимо использовать совмещенное освещение, при котором естественное дополняется искусственным освещением.

#### 5.4.4 Электромагнитное излучение

Электромагнитное излучение - распространяющееся в пространстве возмущение электрических и магнитных полей [25]. Источниками электромагнитного излучения в данном исследовании являются мониторы и системный блок.

Оценка величины уровней ЭМП, проведенная по паспортным данным компьютера и монитора, показала их соответствие нормам ТСО-03 и СанПиН 2.2.2/2.4.1340-03 [26]. В таблице 5.6 приведены нормы уровня ЭМП, которым соответствует техника в кабинете.

Таблица 5.6 – Допустимые уровни ЭМП, создаваемых ПК

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	в диапазоне частот 5 Гц - 2 кГц	25 В/м
	в диапазоне частот 2 кГц - 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот 5 Гц - 2 кГц	250 нТл
	в диапазоне частот 2 кГц - 400 кГц	25 нТл
Электростатический потенциал экрана видеомонитора		500 В

Для того, чтобы снизить воздействие таких видов излучения, рекомендуют применять такие мониторы, у которых уровень излучения понижен (MPR-II, ТСО-92, ТСО-99), а также установить защитные экраны и соблюдать режимы труда и отдыха.

## 5.4 Экологическая безопасность

Экологическая безопасность и охрана окружающей среды являются одними из важнейших факторов при выполнении работ любого характера. При работе в офисном помещении за персональным ПК отсутствуют выбросы в окружающую среду и влияния на жилищную зону.

Поскольку при разработке данной магистерской диссертации использовался компьютер, необходимо помнить о правильной утилизации компьютерного лома после выхода из строя данного ПК. В соответствии с постановлением правительства №340 СанПиН 2.2.2. 542-96 «Гигиенические требования к видеодисплейным терминалам, персональным электронно-вычислительным машинам и организации работы» юридическим лицам запрещено самостоятельно утилизировать компьютерную технику. Необходимо найти организацию, которая занимается утилизацией в частном порядке. Это относится к следующим видам отходов:

- Твердые отходы, относящихся к IV классу опасности (системный блок компьютера, принтеры, сканеры, клавиатура, манипулятор «мышь») [27];
- Жидкие отходы: сточные воды;
- Люминесцентные лампы [28].

## 5.5 Безопасность в чрезвычайных ситуациях

Одними из наиболее вероятных и разрушительных видов чрезвычайных ситуаций являются пожар или взрыв на рабочем месте.

Всякий работник при обнаружении пожара должен [28]:

- незамедлительно сообщить об этом в пожарную охрану;
- принять меры по эвакуации людей, каких-либо материальных ценностей согласно плану эвакуации;
- отключить электроэнергию, приступить к тушению пожара первичными средствами пожаротушения.

При возникновении пожара должна сработать система пожаротушения, передав на пункт пожарной станции сигнал о ЧС. В случае если система не сработала, то необходимо самостоятельно произвести вызов пожарной службы по телефону 101, сообщить точный адрес места возникновения ЧС и ожидать приезда специалистов.

Рабочее место располагается по адресу ул. Мокрушина 9, стр. 16 (Бизнес-центр «Вертикаль») в 332 кабинете. На рис. 5.1 представлен план эвакуации правого крыла третьего этажа бизнес-центра «Вертикаль».

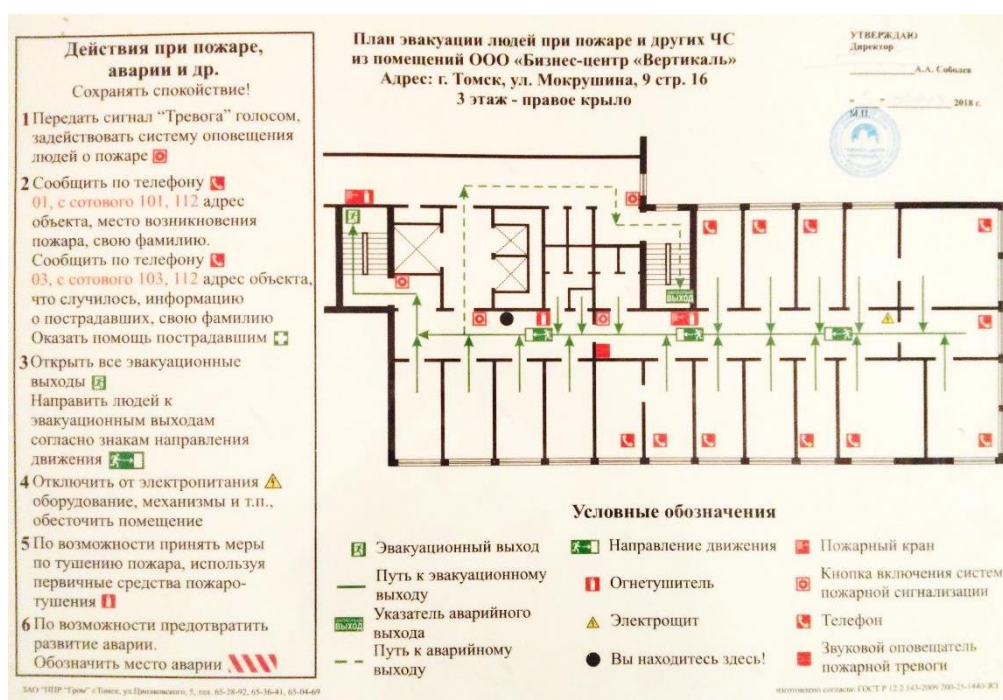


Рисунок 5.1 – План эвакуации при чрезвычайных ситуациях

## **5.6 Вывод по разделу**

Основным фактором, влияющим на производительность людей, занимающиеся разработкой программного обеспечения являются комфортные и безопасные условия труда. Условия труда в рабочем помещении характеризуются возможностью воздействия на работников следующих производственных факторов: действие микроклимата, шума, уровня освещенности, электромагнитного излучения, параметров технологического оборудования и рабочего места. В данном разделе проанализированы вредные и опасные факторы, влияющие на здоровье человека, основные меры их профилактики и ликвидации. Были отмечены источники негативного воздействия, меры коллективной и индивидуальной защиты.

## ЗАКЛЮЧЕНИЕ

Практическим результатом моей работы является разработанный модуль для интеграции информационных систем, XWiki и YouTrack.

В процессе выполнения магистерской диссертации, были изучены проблемы интеграции информационных систем, возможности интеграции вики-системы XWiki посредством расширения функционала системы путем создание модуля расширения.

В ходе выполнения выпускной квалификационной работы, был спроектирован и программно реализован способ интеграции вики-системы XWiki и системы управления задачами YouTrack посредством расширения функционала XWiki взаимодействием с веб-сервисом. Подробнее о взаимодействии веб-сервиса и системы управления задачами YouTrack представлено в работе магистранта гр. 8ИМ71 Евстафьева С.Н.

Разработка модуля расширения для вики-системы XWiki велась на языке программирование Java. При разработке использовалась среда разработки IntelliJ IDEA, и система сборки проектов Maven.

Функциональность разработанного модуля может быть расширена путем добавления возможностей по настройке соединения модуля с веб-сервером. В дальнейшем планируется опубликовать результат работы в открытый доступ на GitHub и ресурсах Openssource российского сообщества разработчиков.

## СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

- 1.«О XWiki» [Электронный ресурс] // <https://habr.com/post/265811/> (Дата обращения 29.01.2019).
- 2.«О XWiki 2 » [Электронный ресурс] // <https://habr.com/post/279101/> (Дата обращения 01.02.2019).
- 3.«Проблемы интеграции» [Электронный ресурс] // <http://ict.informika.ru/vconf/files/10137.pdf> (Дата обращения 17.02.2019).
- 4.«XWiki» [Электронный ресурс] // <http://www.intranetno.ru/tool/524/> (Дата обращения 10.02.2019).
- 5.«Second Generation Wiki» [Электронный ресурс] // <https://www.xwiki.org/xwiki/bin/view/Documentation/UserGuide/Features/SecondGenerationWiki/> (Дата обращения 12.02.2019).
- 6.«XWiki» [Электронный ресурс] // <https://ruprogi.ru/software/xwiki> (Дата обращения 12.02.2019).
- 7.«Спецификация требований программного обеспечения»[Электронный ресурс] // <https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%B> (Дата обращения 29.01.2019).
- 8.«Альтернативы XWiki» [Электронный ресурс] // <https://ruprogi.ru/software/xwiki> (Дата обращения 20.02.2019).
- 9.«Системы сборки проектов» [Электронный ресурс] // <http://www.apache-maven.ru/> (Дата обращения 13.02.2019).
- 10.«Maven» [Электронный ресурс] // <https://habr.com/ru/post/78252/> (Дата обращения 14.02.2019).
- 11.«Полная версия: Руководство по Maven» [Электронный ресурс] // <https://proselyte.net/tutorials/maven/> (Дата обращения 14.02.2019).
- 12.«Руководство IntelliJ IDEA» [Электронный ресурс] // <https://javarush.ru/groups/posts/734-rukovodstvo-poljhzovatelja-intellij-idea-osnovih-i-ustanovka> (Дата обращения 09.02.2019).

13. «Схема базы данных XWiki» [Электронный ресурс] // <https://www.xwiki.org/xwiki/bin/view/Documentation/DevGuide/DatabaseSchema> / (Дата обращения 02.02.2019).
14. «XWiki REST API» [Электронный ресурс] // <https://www.xwiki.org/xwiki/bin/view/Documentation/UserGuide/Features/XWikiRESTfulAPI> (Дата обращения 03.02.2019).
15. «BitLoom» [Электронный ресурс] // [http://blog.fabio.mancinelli.me/2011/03/07/XWikis\\_RESTful\\_API.html](http://blog.fabio.mancinelli.me/2011/03/07/XWikis_RESTful_API.html) (Дата обращения 03.02.2019).
16. «Создание компонентов XWiki» [Электронный ресурс] // <https://www.xwiki.org/xwiki/bin/view/Documentation/DevGuide/Tutorials/WritingComponents/> (Дата обращения 08.02.2019).
17. «Creating an Extension» [Электронный ресурс] // <https://www.xwiki.org/xwiki/bin/view/Documentation/DevGuide/Tutorials/CreatingExtensions/> (Дата обращения 08.02.2019).
18. «XWiki JavaScript API» [Электронный ресурс] // <https://www.xwiki.org/xwiki/bin/view/Documentation/DevGuide/FrontendResources/JavaScriptAPI/> (Дата обращения 08.02.2019).
19. СанПиН 2.2.4.548 – 96. Гигиенические требования к микроклимату производственных помещений. – М.: Информационно-издательский центр Минздрава России, 1997. – 20 с.
20. Борьба с шумом на производстве: Справочник / Е.Я. Юдин, Л.А. Борисов; Под общ. ред. Е.Я. Юдина – М.: Машиностроение, 1985. – 400с.
21. ГОСТ 12.1.003-2014. ССБТ. Шум. Общие требования безопасности.
22. СНиП 23-03-2003. Защита от шума.
23. СНиП 23-05-95\*. Естественное и искусственное освещение. – М.: Центр проектной продукции в строительстве, 2011. – 70 с.
24. СанПиН 2.2.2/2.4.1340-03. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы. – М.: Информационно-издательский центр Минздрава России, 2003. – 54 с.



25. Безопасность жизнедеятельности. /Под ред. Н.А. Белова - М.: Знание, 2000 - 364с.
26. СанПин 2.17.1322-03. Гигиенические требования к размещению и обезвреживанию отходов производства и потребления: Санитарно-эпидемиологические правила и нормативы. – М.: Федеральный центр госсанэпиднадзора Минздрава России, 2004. – 16 с.
27. Постановление Правительства РФ от 25 апреля 2012 года N 390 «О противопожарном режиме» (с изменениями на 7 марта 2019 года).
28. ГОСТ Р 51768-2001 Ресурсосбережение. Обращение с отходами. Методика определения ртути в ртутьсодержащих отходах. Общие требования.

# ПРИЛОЖЕНИЕ А

(обязательное)

## Раздел III

### DEVELOPMENT OF SOFTWARE PRODUCT

Студент

Группа	ФИО	Подпись	Дата
8ИМ71	Дергачев Антон Олегович		

Руководитель ВКР

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мирошниченко Евгений Александрович	к.т.н.		

Консультант отделения ИТ

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИТ	Мирошниченко Евгений Александрович	к.т.н.		

Консультант-лингвист отделения иностранных языков ШБИП

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОИЯ	Сидоренко Татьяна Валерьевна	к.п.н.		

### 3 DEVELOPMENT OF SOFTWARE PRODUCT

The development of a software product must be done for its subsequent implementation after the analysis and design of the integration module of the XWiki wiki system has been performed.

#### 3.1 Interaction of the integration module with the XWiki API

The XWiki wiki system provides access to all elements through the API (application programming interface), based on the semantics of HTTP, RESTful API.

Access to the service by default is via the access point «<http://localhost:8080/xwiki/rest>». So, the /wikis resources on the localhost server on port 8080 can be obtained at the following URL:

«<http://localhost:8080/xwiki/rest/wikis>».

This method of obtaining resources allows to get them in XML format.

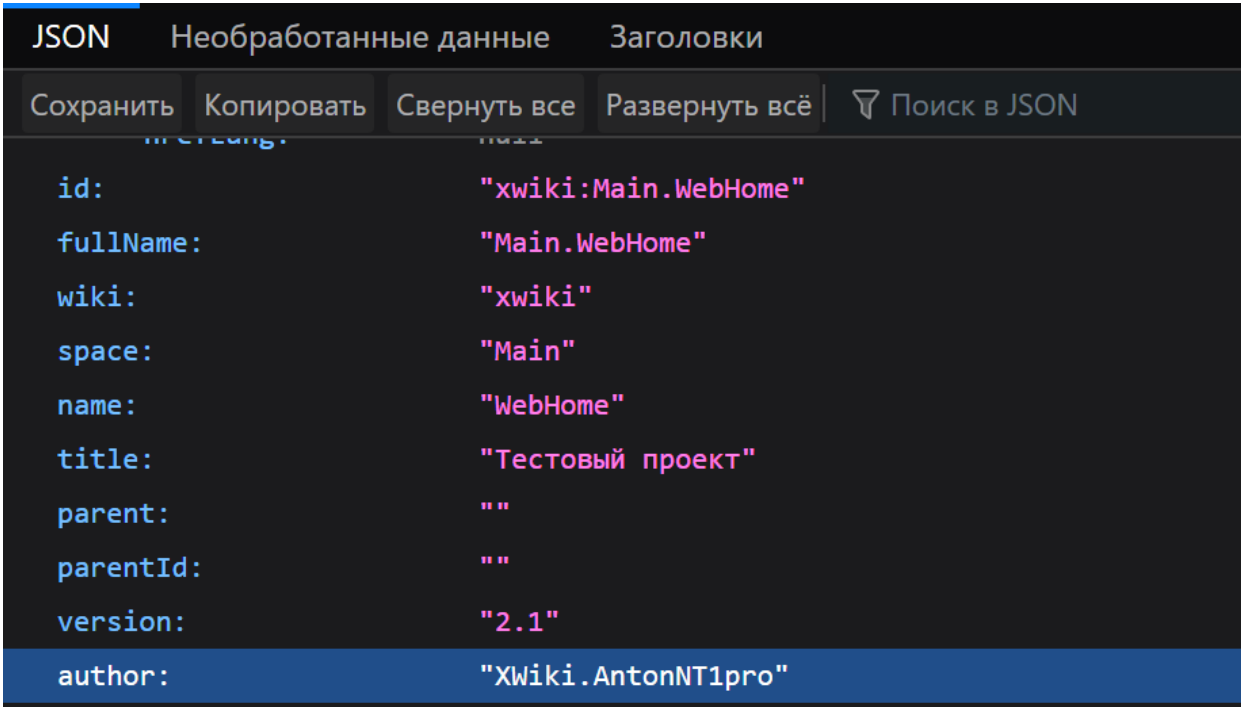
The API XWiki allows to extract it in JSON format in addition to extracting content in XML format:

«<http://localhost:8080/xwiki/rest/wikis/xwiki/spaces/Main/pages/WebHome?media=json>». – The result of the request is presented in Figure 3.1.

```
JSON  Необработанные данные  Заголовки
Сохранить  Копировать  Свернуть все  Развернуть все  🔍 Поиск в JSON
created: 1545826283000
creator: "XWiki.superadmin"
creatorName: null
modified: 1558251649000
modifier: "XWiki.AntonNTipro"
modifierName: null
comment: ""
content: ""= Плагин интеграции XWiki и YouTrack =\n\nВ процессе создания программного обеспечения мы пользуемся различными инструментами.\n\nВ частности спецификации требований мы ведём в XWiki, а для трекинга задач используем YouTrack.\n\nВ целом процесс сбора требований к программному обеспечению и формирование заданий разработчикам выглядит так:\n\n1. Аналитик общается с заказчиком, в wiki фиксирует основные (высокоуровневые) требования к создаваемому ПО.\n1. Аналитик общается с системным архитектором, в wiki детализирует основные требования.\n1. Аналитик согласовывает с заказчиком детализированные требования, вносит в wiki корректировки по результатам согласования требований.\n1. Руководитель проекта совместно с аналитиком «нарезает» задачи из составленной спецификации и вносит их в YouTrack.\n\nПоскольку на текущий момент XWiki и YouTrack не имеют интеграции друг с другом, такой процесс сбора требований и выдачи задач имеет следующие недостатки:\n\n1. При создании задачи в YouTrack приходится вручную копировать описание задачи из XWiki или указывать ссылку на описание задачи в XWiki.\n1. В ходе разработки ПО задачи могут уточняться и детализироваться, при внесении корректировок в XWiki изменения, приходится вручную дублировать в YouTrack.\n1. Сложно определить, все ли задачи из XWiki созданы в YouTrack.\n\nДля решения описанных проблем необходимо разработать плагин для XWiki, реализующий следующие функции:\n\n1. создавать issue в YouTrack из описания задачи в XWiki;\n1. создавать в issue YouTrack ссылку на описание задачи в XWiki;\n1. обновлять описание issue в YouTrack из описания задачи в XWiki;\n\nРабота плагина для пользователя может выглядеть следующим образом:\n\n1. Пользователь в XWiki выделяет текст требования;\n1. Пользователь в XWiki вызывает команду «Создать требование»;\n1. Плагин в YouTrack:\n1. создаёт в issue (поля заполняются по умолчанию);\n1. вносит в поле Description выделенный текст;\n1. добавляет в поле Description ссылку на скопированный текст;\n1. Плагин в XWiki:\n1. выделенный текст оформляет специализированным блоком;\n1. добавляет ссылку на созданное issue в YouTrack/""
class: null
objects: null
attachments: null
```

Figure 3.1 – The result of the request «getting content» in JSON format.

You can access page objects and their properties using RESTful API. The result of the request «getting the author of the article» is presented in Figure 3.2.



```
JSON      Необработанные данные      Заголовки
Сохранить  Копировать  Свернуть все  Развернуть всё  🔍 Поиск в JSON

id: "xwiki:Main.WebHome"
fullName: "Main.WebHome"
wiki: "xwiki"
space: "Main"
name: "WebHome"
title: "Тестовый проект"
parent: ""
parentId: ""
version: "2.1"
author: "XWiki.AntonNT1pro"
```

Figure 3.2 – The result of the request «getting the author of the article» in JSON format.

Pages can have one or more objects. The objects are instances of a class that contains a set of properties. Some objects can be directly represented as first-class objects, such as comments and tags [14].

Availability of resources (sources of information) is an important concept in REST. A global identifier (URI in HTTP) is available for each source of information [15]. Network components (user agents) and application servers that communicate over a standardized interface (HTTP) are used to manipulate these resources. They also exchange views of these resources. The resource layout for the XWiki pages is shown in Figure 3.2.

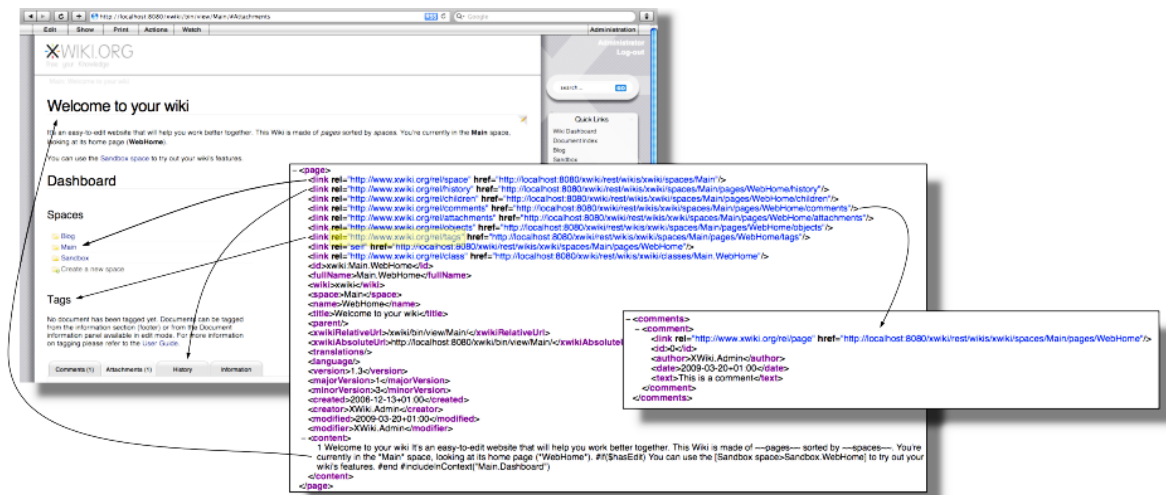


Figure 3.3 – The resource layout for the XWiki pages

Interacting with the XWiki RESTful API: this interface is accessible via HTTP, and you can use any client that can work over HTTP to use this interface. The web browser can also be used to interact with RESTful API.

API-interface XWiki RESTful supports two types of authentication:

- HTTP BASIC Auth – providing your credentials using HTTP.
- XWiki Session – cookie files are used provided by the authentication mechanism. Authentication in the XWiki RESTful API will also be passed.

The HTTP authentication type BASIC Auth was used in the development of the integration module of the XWiki wiki system as part of its pre-diploma practice.

To interact with the RESTful API, XWiki resources were imported into a Java project:

```
import org.xwiki.component.annotation.Component;
import org.xwiki.rest.XWikiResource;
```

### 3.2 The XWiki data model

It is worth considering the wiki data model of the XWiki system, in order to use the available and open XWiki resources through the RESTful API [13]. It is presented in Figure 3.4.

In fact, there is an almost one-to-one correspondence between open resources and entities of this model.

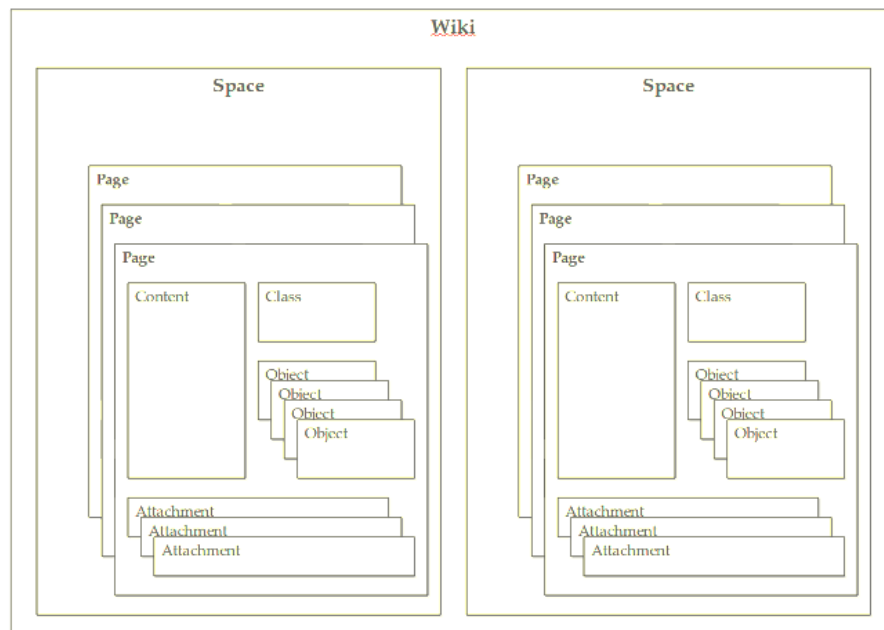


Figure 3.4 – The XWiki wiki system data model diagram

The page is the central entity. The page has associated content and a set of metadata that provides additional information (for example, title, author, creation date, change date, etc.). Page content is usually written using XWiki markup and may contain scripts. These scripts can access the base platform to retrieve and display data stored in the system.

Three types of entities can be associated with the page:

- Attachments are binary objects associated with the page.
- Classes are metadata that defines structure. The class provides the definition of several properties. Each property has a type and several attributes that are used to restrict the values assigned to this property, as well as restrictions on the way it is edited.
- Objects represent structured data according to a scheme defined by a class. Using objects, you can structure data and access it.

Classes and objects make XWiki a powerful and more flexible platform when compared to other peers. Data stored using objects can be edited, searched, manipulated and displayed in a specific way. This is due to their well-defined

structure. Applications can use this mechanism to define, store and edit domain data using the XWiki platform.

A complete map of the resources available through the API RESTful is presented in Application A, B. The diagram shows how from an entry point you can navigate through all the resources provided by the API RESTful [17].

Nodes are URI patterns that specify the address of a resource set. The URI pattern is a parametric URI that defines the structure for addressing a collection of resources. Ribs are connections between representations of various resources.

The complete database scheme is presented in Application C and D.

### **3.3 Developed methods for the integration module**

Methods:

- `getRangeText(int: x1,x2)` takes the positions of the beginning and end letters in the selected text, returns the selected text using the Range object, and passes it to the `getConstruct (String: inStr)` method.
- `createPage(String: header, content)` is a page creation method. It calls the RESTful API XWiki method to create the page, accepts the result of the execution of the `getConstruct()` method, «header» and «content».
- `getPage()` is a method for retrieving XWiki page data. This method uses the RESTful API XWiki «`xwiki/rest/wikis/xwiki/spaces/Main/pages/WebHome?media=json`» to retrieve page data in JSON format. This parses it and passes the value of the «content» key to the `getConstruct` method (String: str).
- `getConstruct(String: str)` – the method splits the received text string into two parts, «header» and «content», then calls the `getHeaderIndex()` method, concatenates the result of the work with the “header” methods, and passes the values to the `sendJSON()` method.

- `getUserName()` is a method for getting user data. This method uses the RESTful API XWiki to retrieve user data in JSON format, parses it, and passes the value of the «User» key to the `sendJSON()` method.

- `getKey(String: headerStr, contentStr)` – the method concatenates the received data «header» + random generated number from 100 to 1000 (`getHeaderIndex()`), and «content», and encrypts the resulting string. The resulting key is used in the `sendJSON()` method.

- `getDeletePage(int: idPage)` is request page removal method. The method sends data in JSON format to a web service indicating the type of the «delete» operation.

- `sendJSON(String: header, content, user, key, url)` – the method sends data in JSON format, the code of this method:

```
{
HttpPost request = new HttpPost("http://10.192.92.17/api/post/create");
StringEntity params =new
StringEntity(«details={\"neader\":header, \"content\":content, \"user\":user,
\"key\":key, \"url\":url,}\"");
request.addHeader("content-type", "application/x-www-form-urlencoded");
request.setEntity(params);
HttpResponse response = httpClient.execute(request);
}
```

«Header» contains the title of the page with the requirement.

«Content» contains a descriptive part of the requirement.

«User» contains the name of the author of the article with the requirement.

«Key» contains an encrypted string.

«Url» contains a link to the page with the requirement.



The request refers to the web service API at the address «http: // localhost / api / post / create» when creating a page with a request.

The request refers to the web service API at «http://localhost /api/post/update» when updating the request page.

A request is made to the web service API at «http://localhost /api/post/delete» when deleting the request page.

### **3.4 Connection module in XWiki**

The integration of the module into the wiki system of XWiki can be done in two ways:

- Copying the developed jar files into the directory intended for XWiki system wiki extensions. The module is registered by recording it in the xwiki.cfg file located in the WEB-INF directory.
- Use the XWiki wiki system for deploying extensions at extensions.xwiki.org. Registration, descriptive part of the developed module and screenshots of its work are required for publishing the extension [16].

The extension to the XWiki wiki was integrated by copying the developed files into the XWiki directory as part of this master's work.

### **3.5 Result of development**

The result of the section "Software Product Development" is a developed integration module for the XWiki wiki system. The module allows you to create requirements pages and send the selected data to the web service about creating, updating, deleting request pages in the XWiki project. The module operation is presented in Figures 3.6 – 3.10.

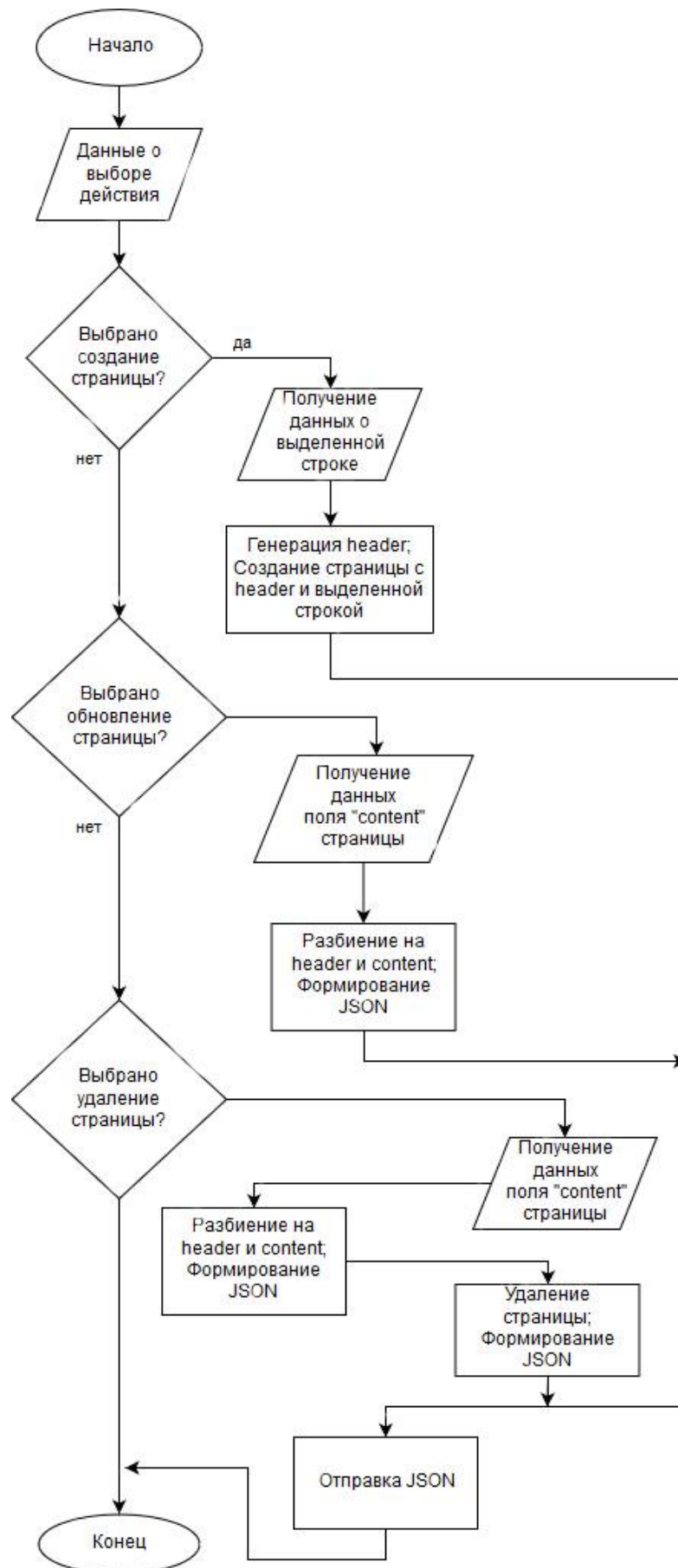


Figure 3.5 – Algorithm of working with the developed module

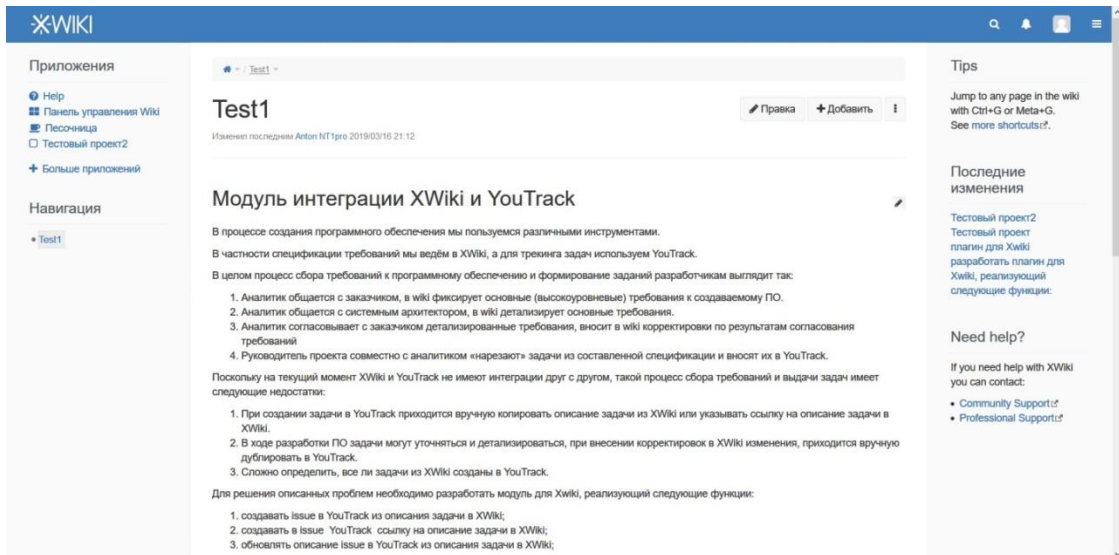


Figure 3.6 – The XWiki Project Home Page

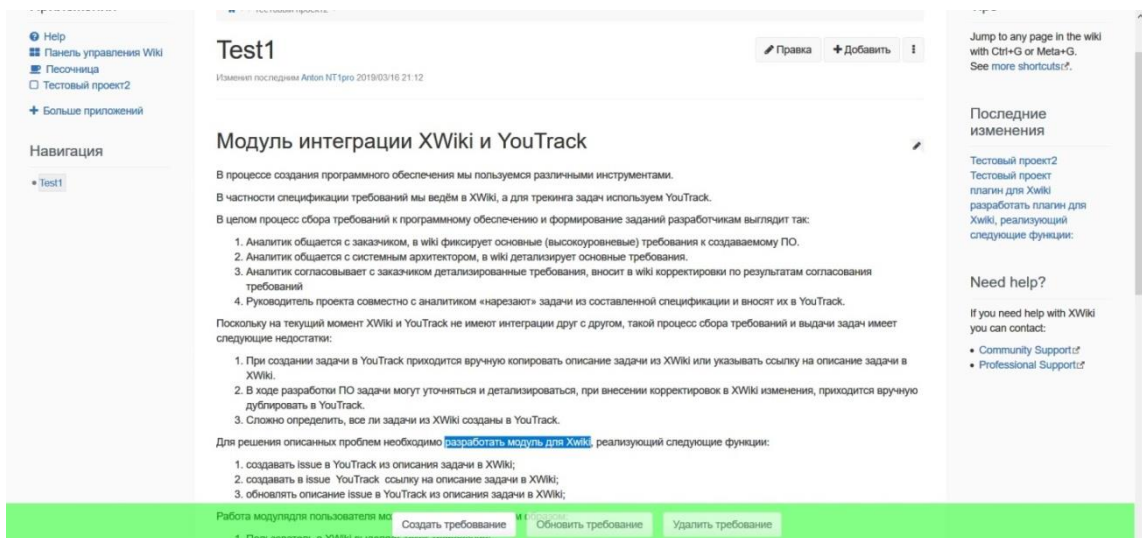


Figure 3.7 – The pop-up action bar

The pop-up action bar becomes available to the user when text is selected on the project requirements pages and allows you to create a requirements page.

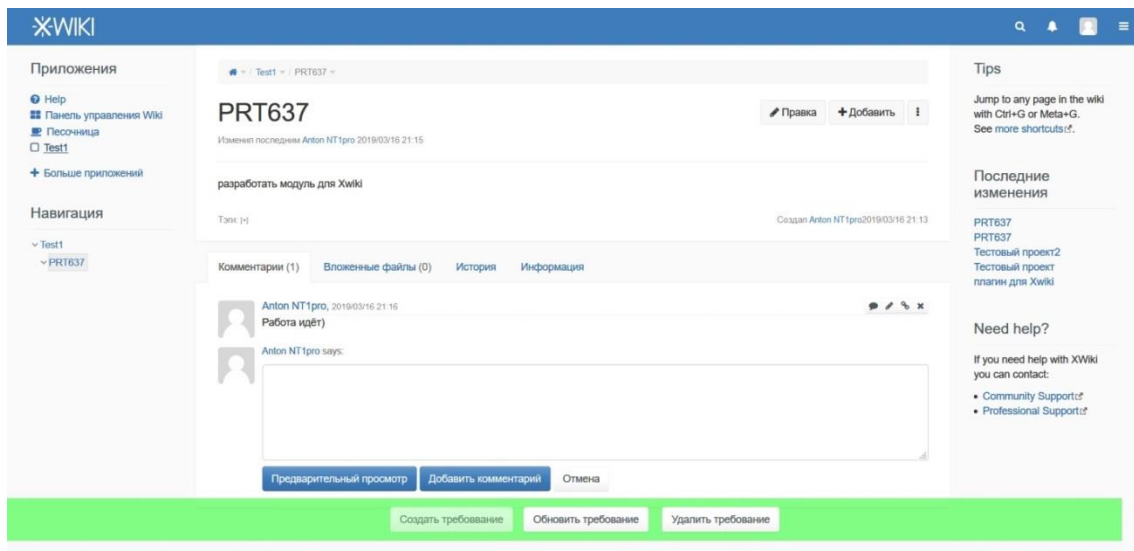


Figure 3.8 – The created page with the requirement.

JSON was sent to the web service when creating the request page:

```
{
  "header": "PRT872",
  "content»: "to create an issue in YouTrack from the task description in
  XWiki",
  "url": "https://localhost:8080/xwiki/bin/view/Test1/PRT872",
  "user": "AntonNT1pro",
  "key":
  "UFJUODcy0YHQvtC30LTQsNCy0LDRgtGMIGlzc3VIINCyIFlvdVRyY
  WNrINC40Lcg0L7Qv9C40YHQsNC90LjRjyDQt9Cw0LTQsNGH0Lgg0
  LIgWFdpa2k=/aHR0cHM6Ly9udDE6ODA4MC94d2lraS9iaW4vdmlldy9
  UZxN0MS9QUIQ4NzI="}
```

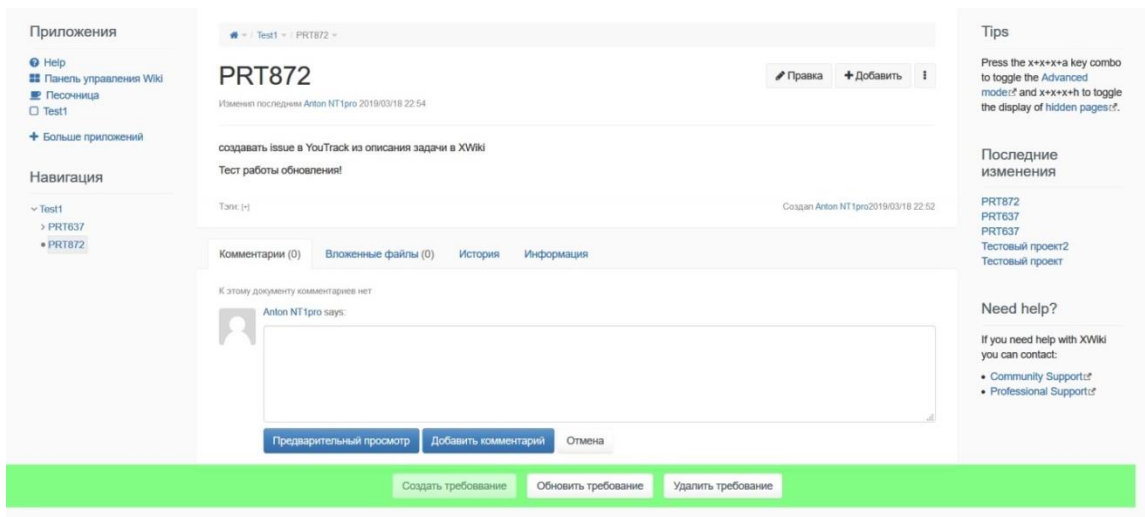


Figure 3.9 – Updating requirements on pages

JSON was sent to the web service when the request was updated:

```
{
  "header": "PRT872",
  "content": "to create an issue in YouTrack from the task description in
  XWiki The test of work update!",
  "url": "https://localhost:8080/xwiki/bin/view/Test1/PRT872",
  "user": "AntonNT1pro",
  "key": "UFJUODcy0YHQvtC30LTQsNCy0LDRgtGMIGlzc3VIINC
  yIFlvdVRyYWNrINC40Lcg0L7Qv9C40YHQsNC90LjRjyDQt9Cw0LTQs
  NGH0Lgg0LIgWFdpa2kg0KLQtdGB0YIlg0YDQsNCx0L7RgtGLINC+0L
  HQvdC+0LLQu9C10L3QuNGPIQ==/aHR0cHM6Ly9udDE6ODA4MC94
  d2lraS9iaW4vdmllldy9UZjR0MS9QUiQ4NzI="}
```

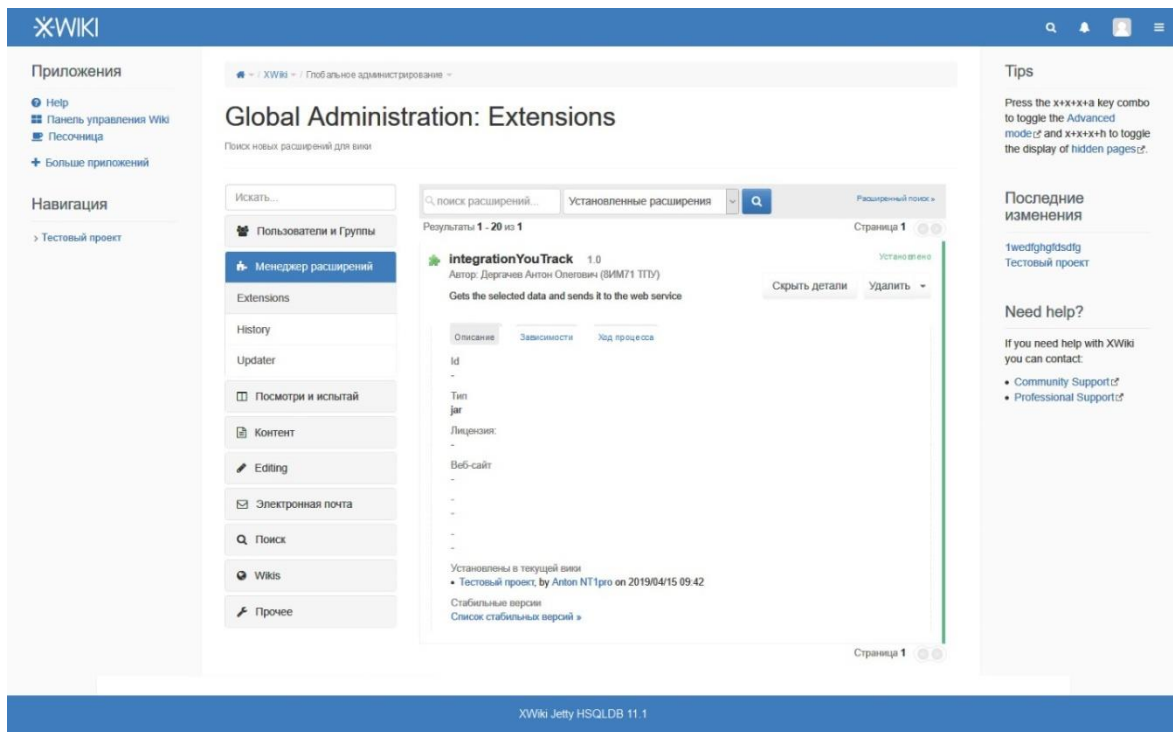
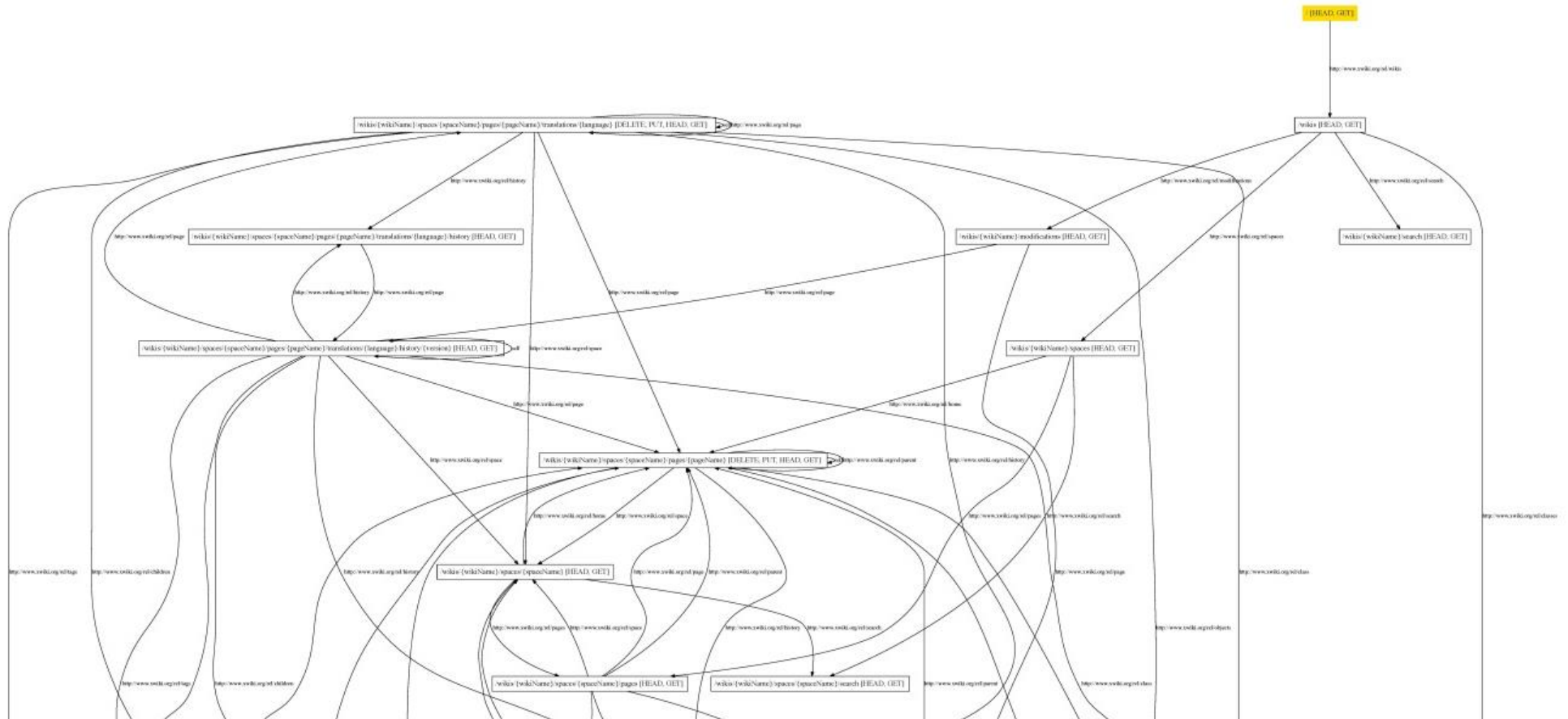


Figure 3.10 – The XWiki Extension bar.

The XWiki wiki extensions bar (module) shows the connected extensions. Information on the developed extension (author of the extension, name of the established project, its author, type of extension) is presented in the figure as part of the implementation of the pre-diploma practice.

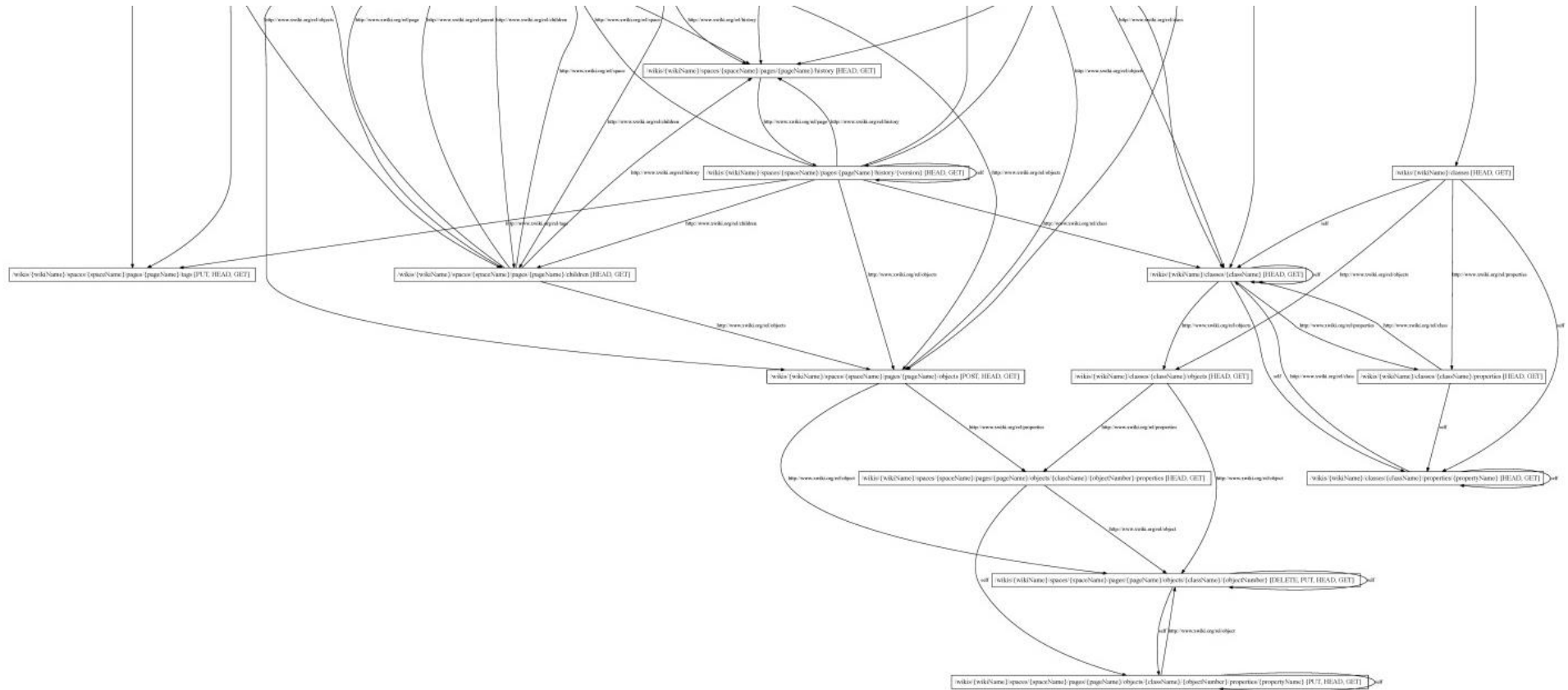
## ПРИЛОЖЕНИЕ Б

### Полная карта ресурсов XWiki, доступных через API RESTful (часть 1)



## ПРИЛОЖЕНИЕ В

### Полная карта ресурсов XWiki, доступных через API RESTful (часть 2)





# ПРИЛОЖЕНИЕ Г

## Полная схема базы данных вики системы XWiki (часть 1)

The image displays a screenshot of a database schema tool showing 14 tables from the XWiki database. Each table is represented by a blue-bordered window with a title bar and a list of columns with their data types. The tables and their columns are:

- xwikidoc**: XWD\_ID BIGINT(20), XWD\_FULLNAME VARCHAR(255), XWD\_NAME VARCHAR(255), XWD\_TITLE VARCHAR(255), XWD\_LANGUAGE VARCHAR(5), XWD\_DEFAULT\_LANGUAGE VARCHAR(5), XWD\_TRANSLATION INT(11), XWD\_DATE DATETIME, XWD\_CONTENT\_UPDATE\_DATE DATETIME, XWD\_CREATION\_DATE DATETIME, XWD\_AUTHOR VARCHAR(255), XWD\_CONTENT\_AUTHOR VARCHAR(255), XWD\_CREATOR VARCHAR(255), XWD\_WEB VARCHAR(255), XWD\_CONTENT LONGTEXT, XWD\_VERSION VARCHAR(255), XWD\_CUSTOM\_CLASS VARCHAR(255), XWD\_PARENT VARCHAR(511), XWD\_CLASS\_XML LONGTEXT, XWD\_ELEMENTS INT(11), XWD\_DEFAULT\_TEMPLATE VARCHAR(255), XWD\_VALIDATION\_SCRIPT VARCHAR(255), XWD\_COMMENT VARCHAR(1023), XWD\_MINOREDIT BIT(1), XWD\_SYNTAX\_ID VARCHAR(50), XWD\_HIDDEN BIT(1).
- xwikircs**: XWR\_DOCID BIGINT(20), XWR\_VERSION1 INT(11), XWR\_VERSION2 INT(11), XWR\_DATE DATETIME, XWR\_COMMENT VARCHAR(1023), XWR\_AUTHOR VARCHAR(255), XWR\_ISDIFF BIT(1), XWR\_PATCH LONGTEXT.
- xwikirecyclebin**: XDD\_ID BIGINT(20), XDD\_FULLNAME VARCHAR(255), XDD\_LANGUAGE VARCHAR(5), XDD\_DATE DATETIME, XDD\_DELETER VARCHAR(255), XDD\_XMLSTORE VARCHAR(50), XDD\_XML LONGTEXT, XDD\_BATCH\_ID VARCHAR(255).
- xwikiattachment**: XWA\_ID BIGINT(20), XWA\_DOC\_ID BIGINT(20), XWA\_FILENAME VARCHAR(255), XWA\_LONGSIZE BIGINT(20), XWA\_SIZE INT(11), XWA\_MIMETYPE VARCHAR(255), XWA\_DATE DATETIME, XWA\_AUTHOR VARCHAR(255), XWA\_COMMENT VARCHAR(1023), XDD\_CONTENTSTORE VARCHAR(50), XDD\_ARCHIVESTORE VARCHAR(50), XDD\_CHARSET VARCHAR(40).
- xwikispace**: XWS\_ID BIGINT(20), XWS\_REFERENCE VARCHAR(255), XWS\_NAME VARCHAR(255), XWS\_PARENT VARCHAR(255), XWS\_HIDDEN BIT(1).
- xwikiattachment\_content**: XWA\_ID BIGINT(20), XWA\_CONTENT LONGBLOB.
- xwikiattachment\_archive**: XWA\_ID BIGINT(20), XWA\_ARCHIVE LONGBLOB.
- xwikilinks**: XWL\_DOC\_ID BIGINT(20), XWL\_LINK VARCHAR(255), XWL\_FULLNAME VARCHAR(255).
- xwikidbversion**: XWV\_VERSION INT(11).
- xwikiid**: XWI\_ID VARCHAR(255).
- activitystream\_events**: ase\_eventid VARCHAR(48), ase\_requestid VARCHAR(48), ase\_stream VARCHAR(255), ase\_date DATETIME, ase\_priority INT(11), ase\_type VARCHAR(255), ase\_application VARCHAR(255), ase\_user VARCHAR(255), ase\_wiki VARCHAR(255), ase\_space VARCHAR(255), ase\_page VARCHAR(255), ase\_hidden BIT(1), ase\_url VARCHAR(2000), ase\_title VARCHAR(2000), ase\_body VARCHAR(2000), ase\_version VARCHAR(30), ase\_param1 VARCHAR(2000), ase\_param2 VARCHAR(2000), ase\_param3 VARCHAR(2000), ase\_param4 VARCHAR(2000), ase\_param5 VARCHAR(2000).
- activitystream\_events\_targets**: eventid VARCHAR(48), value VARCHAR(255).
- activitystream\_events\_status**: ases\_eventid VARCHAR(48), ases\_entityid VARCHAR(255), ases\_read BIT(1).

## ПРИЛОЖЕНИЕ Д

Полная схема базы данных вики системы XWiki (часть 2)

