

Документ подписан простой электронной подписью Информация о владельце: ФИО: Таскаев Сергей Валерьевич Должность: Ректор Дата подписания: 18.03.2025 14:53:17 Уникальный программный ключ: 04c19ed8bfb98f3b6c77a486b8a8788b8722727	МИНОВЕРНАУКИ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)	Рабочая программа дисциплины "Анализ требований и проектирование систем искусственного интеллекта" по направлению подготовки (специальности) 01.03.02 "Прикладная математика и информатика" направленности (профиль) Прикладная математика и искусственный интеллект ФГБОУ ВО «ЧелГУ»	стр. 1
---	--	---	--------

Рабочая программа дисциплины (модуля)*

Анализ требований и проектирование систем искусственного интеллекта

Направление подготовки (специальность)

01.03.02 Прикладная математика и информатика

Направленность (профиль)

Прикладная математика и искусственный интеллект

Присваиваемая квалификация (степень)

бакалавр

Форма обучения

очная

Год набора 2024

*Рабочая программа дисциплины (модуля) адаптирована для инклюзивного обучения инвалидов и лиц с ограниченными возможностями здоровья

Челябинск 2023 г.



Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)
4. Объем дисциплины (модуля)
5. Структура и содержание дисциплины (модуля)
6. Фонд оценочных средств
 - 6.1. Перечень видов оценочных средств
 - 6.2. Типовые контрольные задания и иные материалы для текущей аттестации
 - 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации
 - 6.4. Критерии оценивания
7. Учебно-методическое и информационное обеспечение дисциплины (модуля)
 - 7.1. Рекомендуемая литература
 - 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"
 - 7.3. Перечень информационных технологий
8. Материально-техническое обеспечение дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Специальные условия освоения дисциплины обучающимися с инвалидностью и ограниченными возможностями здоровья



1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Преподаваемая дисциплина посвящена изучению двух важных этапов жизненного цикла программного обеспечения - анализу предметной области и проектированию архитектуры для систем искусственного интеллекта. Преподавание и изучение дисциплины следует рассматривать как полезную составляющую профессиональной подготовки. Целью преподавания дисциплины является обучение студентов методам выявления, анализа и разработки требований и методам проектирования программных систем с использованием искусственного интеллекта. Задачи дисциплины заключаются в том, чтобы студенты овладели навыками анализа предметной области, создания и описания объектно-ориентированных моделей предметной области, выполнения системного анализа и разработки на его основе архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения, систем с использованием искусственного интеллекта; навыками создания спецификаций, как для всей системы в целом, так для отдельных подсистем и модулей.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Цикл (раздел) ОПОП: Б1.О.37

2.1 Требования к предварительной подготовке обучающегося:

1. Основы защиты данных в интеллектуальных системах:

Знает: [УК-1.2. 3-1.] цели задачи и предмет, основные понятия информационной безопасности, информационные угрозы, их классификацию, возможные последствия для организаций различных форм собственности и критерии оценки защищённости информационных систем и систем искусственного интеллекта Умеет: [УК-1.2. У-2.] сознавать опасности и угрозы, возникающие в профессиональной деятельности и в социальной сфере, соблюдать основные требования информационной безопасности, в том числе защиты государственной тайны; [УК-1.2. У-3.] работать с информацией с учётом требований информационной безопасности Имеет практический опыт

2. Компьютерные сети:

Знает: [УК-1.1. 3-4.] основные методы оценки экономической эффективности применяемого программного и аппаратного обеспечения Умеет: [УК-1.1. У-3.] формировать и использовать критерии оценки эффективности применения программного и аппаратного обеспечения в профессиональной деятельности Имеет практический опыт

3. Объектно-ориентированное программирование:

Знает: [ПК-2.2. 3-1.] современные языки программирования, библиотеки и программные платформы для объектно-ориентированного программирования приложений систем интеллекта, методику разработки программ с использованием технологии объектноориентированного программирования, синтаксис языка объектно-ориентированного программирования C++; устройство и принципы построения объектно-ориентированных библиотек Умеет: реализовывать программно и использовать на практике математические алгоритмы, с применением высокоуровневого языка программирования C++, адаптировать и использовать шаблоны объектноориентированного программирования для решения профессиональных задач Имеет практический опыт: разработки компьютерных программ на языке C++, применения объектных технологий разработки программных систем

4. Современные технологии разработки программных систем искусственного интеллекта:

Знает: [УК-1.1. 3-3.] современное состояние информационно-коммуникационных технологий в мире и перспективы их развития, принципы работы современных информационных технологий, [ОПК-1.1. 3-1.] рынок информационных систем и информационнокоммуникационных технологий, автоматизирующих организационно-технические и экономические процессы, [ПК-2.1. 3-1.] основные программные платформы и компоненты систем искусственного интеллекта: механизмы логического вывода (рассуждений), объяснений, приобретения знаний, интеллектуальных интерфейсов, принципы Data Ops и Dev Ops, современные компьютерные технологии разработки программных систем Умеет: использовать современные информационные технологии для решения задач профессиональной деятельности, [ПК-2.1. У-1.] настраивать основные программные платформы и компоненты систем искусственного интеллекта: механизмов логического вывода (рассуждений), объяснений, приобретения знаний, интеллектуальных интерфейсов на особенности проблемной области, участвует в их разработке Имеет практический опыт: применения современных информационных технологий при проектировании систем искусственного интеллекта, участия в разработке прикладного программного обеспечения.

Компьютерные сети

Современные технологии разработки программных систем искусственного интеллекта



Рабочая программа дисциплины "Анализ требований и проектирование систем искусственного интеллекта" по направлению подготовки (специальности) 01.03.02 "Прикладная математика и информатика" направленности (профилю) Прикладная математика и искусственный интеллект ФГБОУ ВО «ЧелГУ»

стр. 4

Объектно-ориентированное программирование

Основы защиты данных в интеллектуальных системах

2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

Технологии и модели управления проектами в интеллектуальных системах

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

ПК-3: Способен классифицировать и идентифицировать задачи искусственного интеллекта, выбирать адекватные методы и инструментальные средства решения задач искусственного интеллекта

Уметь:

[ПК-1.2. У-1.] осуществлять оценку критериев выбора методов и инструментальных средств решения задач с помощью систем искусственного интеллекта и выбор методов и инструментальных средств в зависимости от особенностей проблемной и предметной областей

УК-91: Способен планировать и организовывать свою деятельность в цифровом пространстве с учётом правовых и этических норм взаимодействия человека и искусственного интеллекта и требований информационной безопасности

Знать:

[УК-1.3. 3-2.] международные и национальные стандарты в области искусственного интеллекта и смежных областях

Уметь:

[УК-1.3. У-2.] применять международные и национальные стандарты в области искусственного интеллекта и смежных областях для решения задач в профессиональной деятельности

ОПК-91: Способен анализировать, разрабатывать, внедрять и выполнять организационно технические и экономические процессы с применением технологий и систем искусственного интеллекта

Уметь:

[ОПК-1.1. У-1.] выбирать рациональные решения в области информационных технологий и систем искусственного интеллекта при построении организационно-технических и экономических процессов

Владеть:

Имеет практический опыт: нахождения рациональных решений в области информационных технологий и систем искусственного интеллекта при построении организационно-технических и экономических процессов

ПК-4: Способен разрабатывать и тестировать программные компоненты решения задач в системах искусственного интеллекта

Знать:

[ПК-2.3. 3-1.] основные критерии качества систем искусственного интеллекта, методы и инструментальные средства тестирования и качества функционирования систем искусственного интеллекта

Уметь:

[ПК-2.3. У-1.] проводить тестирование работоспособности и качества функционирования систем искусственного интеллекта и проверять выполнение требований к системам искусственного интеллекта со стороны пользователя

Владеть:

Имеет практический опыт: тестирования работоспособности и качества функционирования систем искусственного интеллекта

ПК-5: Способен разрабатывать и применять методы машинного обучения для решения задач

Знать:

[ПК-4.2. 3-1.] методы и критерии оценки качества моделей машинного обучения

Уметь:

[ПК-4.2. У-1.] определять критерии и метрики оценки результатов моделирования при построении систем искусственного интеллекта в исследуемой области

Владеть:

Имеет практический опыт: оценки качества моделей машинного обучения



В результате освоения дисциплины обучающийся должен

3.1 Знать:	
3.1.1	1. [ПК-4.2. 3-1.] методы и критерии оценки качества моделей машинного обучения
3.1.2	2. [ПК-2.3. 3-1.] основные критерии качества систем искусственного интеллекта, методы и инструментальные средства тестирования и качества функционирования систем искусственного интеллекта
3.1.3	3. [УК-1.3. 3-2.] международные и национальные стандарты в области искусственного интеллекта и смежных областях
3.2 Уметь:	
3.2.1	1. [ПК-4.2. У-1.] определять критерии и метрики оценки результатов моделирования при построении систем искусственного интеллекта в исследуемой области
3.2.2	2. [ПК-2.3. У-1.] проводить тестирование работоспособности и качества функционирования систем искусственного интеллекта и проверять выполнение требований к системам искусственного интеллекта со стороны пользователя
3.2.3	3. [ПК-1.2. У-1.] осуществлять оценку критериев выбора методов и инструментальных средств решения задач с помощью систем искусственного интеллекта и выбор методов и инструментальных средств в зависимости от особенностей проблемной и предметной областей
3.2.4	4. [ОПК-1.1. У-1.] выбирать рациональные решения в области информационных технологий и систем искусственного интеллекта при построении организационно-технических и экономических процессов
3.2.5	5. Умеет: [УК-1.3. У-2.] применять международные и национальные стандарты в области искусственного интеллекта и смежных областях для решения задач в профессиональной деятельности
3.3 Владеть:	
3.3.1	1 оценки качества моделей машинного обучения
3.3.2	2. тестирования работоспособности и качества функционирования систем искусственного интеллекта
3.3.3	3. нахождения рациональных решений в области

4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость	3 ЗЕТ
Часов по учебному плану : 108 в том числе : аудиторные занятия : 64 самостоятельная работа : 35,5 : контактная работа: 72,5 ИКР: 0	Виды контроля в семестрах: экзамены 6

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Литература
	Раздел 1. Разработка и анализ требований			
1.1	Основы управления требованиями к программному обеспечению. Процессы инженерии требований: выявление требований, спецификация, анализ и управление. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.2	Типы требований: функциональные, нефункциональные, атрибуты качества. Потребности и функции продукта. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.3	Выявление задач автоматизации с помощью методов искусственного интеллекта и машинного обучения. Ограничения методов ИИ и МО. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3



Рабочая программа дисциплины "Анализ требований и проектирование систем искусственного интеллекта" по направлению подготовки (специальности) 01.03.02 "Прикладная математика и информатика" направленности (профилю) Прикладная математика и искусственный интеллект ФГБОУ ВО «ЧелГУ»				стр. 6
1.4	Выявление требований: определение потребностей, целей и требований. Заказчики и другие заинтересованные лица. Интервью, наблюдения, совещания, мозговой штурм, раскадровки, прототипирование. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.5	Оценивание пригодности исходных данных для экспертных систем и машинного обучения, эффекта от использования результатов /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.6	Спецификация требований: текстовые и графические нотации и языки (UML, нотации пользовательских требований). Методы написания высококачественных требований. Стандарты документирования. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.7	Анализ требований: инспекция, аттестация, завершенность, обнаружение конфликтов и несоответствий. Анализ взаимодействия элементов функциональности и разрешение противоречий. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.8	Требования к системам искусственного интеллекта и к системам с использованием методов ИИ: встроенным системам, потребительским системам, web-системам, бизнес-системам, научным системам и другим инженерным системам. Согласование требований и управление рисками. Интеграция анализа требований и процессов разработки программного обеспечения (включая Agile- процессы). /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.9	Анализ множества существующих программных систем: измерение качества и восстановление требований (reverse engineering) по программе. Использование инструментов управления требованиями. /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.10	Анализ проблемы. Выявление причин проблемы. /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.11	Моделирование, прототипирование и анализ требований с помощью средств UML /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.12	Выбор и оценка задач для автоматизации с помощью методов искусственного интеллекта и машинного обучения. /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.13	Оценивание пригодности исходных данных, эффекта от использования результатов /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.14	Создание документа-концепции для учебного проекта /Пр/	6	4	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.15	Создание спецификации для учебного проекта /Пр/	6	4	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
1.16	Подготовка к практическим занятиям /Ср/	6	10	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
Раздел 2. Проектирование архитектуры и разработка спецификаций модулей				
2.1	Проблемы разработки сложных программ. Структурный подход в проектировании ПО и классификация структурных методологий. Диаграммы «сущность-связь» (ERD), диаграммы потоков данных (DFD), SADT-модели (стандарт IDEF0). /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.2	Архитектура ПО, влияние архитектуры на свойства ПО. Особенности разработки сложных программных систем: иерархичность, групповая разработка, сборочное проектирование. Преимущества и недостатки объектно-ориентированного подхода. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3



Рабочая программа дисциплины "Анализ требований и проектирование систем искусственного интеллекта" по направлению подготовки (специальности) 01.03.02 "Прикладная математика и информатика" направленности (профилю) Прикладная математика и искусственный интеллект ФГБОУ ВО «ЧелГУ»				стр. 7
2.3	Основные понятия унифицированного языка моделирования (UML). Диаграммы прецедентов, диаграммы классов, диаграммы взаимодействий, диаграммы последовательности действий, диаграммы состояний, компонентные диаграммы. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.4	Углубленное изучение проектирования программного обеспечения, шаблонов проектирования, сред разработки и архитектур. Применения шаблонов проектирования к прикладным задачам. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.5	Классификация методов искусственного интеллекта и машинного обучения. Способы интеграция компонент ИИ в приложение. /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.6	Паттерны проектирования для интеллектуальных систем /Лек/	6	4	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.7	Классификация CASE-систем и их сравнительная характеристика. CASEсредства для разработки интеллектуальных систем /Лек/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.8	Разработка архитектуры учебного проекта /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.9	Разработка диаграммы классов для учебного проекта /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.10	Создание спецификации для нескольких модулей учебного проекта /Пр/	6	2	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.11	Разработка диаграмм взаимодействий, диаграмм последовательности действий для учебного проекта /Пр/	6	4	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.12	Детальное проектирование компонент интеллектуальных систем с использованием паттернов проектирования. /Пр/	6	4	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
2.13	Подготовка к практическим занятиям /Ср/	6	16,5	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
Раздел 3. Иная контрольная работа				
3.1	Индивидуальные консультации, текущий контроль /КурсР/	6	8,5	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3
Раздел 4. Экзамен				
4.1	Подготовка к экзамену /Ср/	6	9	Л1.1 Л1.2 Л1.3 Л1.4Л2.1 Л2.2 Э1 Э2 Э3

6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

6.1. Перечень видов оценочных средств

Текущий контроль: 12 практических работ, каждая оценивается отдельно по шкале от 0 до 10 баллов.
Активность на занятиях, посещаемость 100% посещение (допускаются пропуски по уважительной причине) - 3 балла
85-99% посещение - 2 балла
Работа у доски 1 балл за решение задачи у доски, но не более 5 баллов за семестр
Экзамен: билеты 2 вопроса - максимальный балл - 10
Экзамен проводится в форме письменного ответа по билету и собеседования. После выдачи билета студенту предоставляется 1 час на подготовку ответа по теоретическим вопросам.

6.2. Типовые контрольные задания и иные материалы для текущей аттестации

Смотри приложение для практических занятий.



6.3. Типовые контрольные вопросы и задания для промежуточной аттестации

Вопросы к экзамену

1. Определения: проблема, потребность, функция (feature), требования к ПО (функциональные, нефункциональные)
 2. Анализ проблемы. Этапы анализа проблемы
 3. Типы требований
 4. Системный анализ при разработке ПО. Производные требования.
 5. Преграды на пути выявления требований
 6. Список способов выявления потребностей и требований (Лэффенгуэлл, Вигерс)
 7. Интервьюирование (задачи, правила)
 8. Совещание (задачи, правила)
 9. Мозговой штурм (задачи, правила)
 10. Раскадровки (задачи, правила)
 11. Роль прототипирования в выявлении требований. Виды прототипирования ПО.
 12. Диаграмма вариантов использования (задачи, обозначения)
 13. Обыгрывание ролей (задачи, правила)
 14. Выявление задач автоматизации с помощью методов искусственного интеллекта и машинного обучения. Ограничения методов ИИ и МО.
 15. Оценка пригодности исходных данных для экспертных систем и машинного обучения, эффекта от использования результатов
 16. Документ-концепция, спецификация требований к ПО (назначение, рекомендации)
 17. Способы описания требований, проверка качества описания на естественном языке.
 18. Управление масштабом (задачи, методы)
 19. Управление требованиями (задачи, методы)
 20. Диаграмма развертывания (назначение, обозначения)
 21. Диаграмма классов (назначение, обозначения)
 22. Диаграмма компонентов (назначение, обозначения)
 23. Диаграмма объектов (назначение, обозначения)
 24. Диаграмма пакетов (назначение, обозначения)
 25. Диаграмма деятельности (назначение, обозначения)
 26. Диаграмма состояний (назначение, обозначения)
 27. Диаграмма последовательности (назначение, обозначения)
 28. Диаграмма коммуникации (назначение, обозначения)
 29. Влияние парадигм на архитектуру (ФП, ООП, СП)
 30. Принципы SOLID
 31. Принципы для связности компонентов (REP, CCP, CRP)
 32. Принципы для сочетаемости компонентов (ADP, SDP, SAP)
 33. Цель архитектуры, влияние на разработку, развертывание, эффективность работы, сопровождение.
 34. Разделение архитектуры на горизонтальные и вертикальные уровни (используемые принципы). Границы системы, трансграничные вызовы и информация о структуре обрабатываемых данных.
 35. Рекомендуемая "чистая архитектура" (Мартин), правило зависимостей.
 36. Классификация методов искусственного интеллекта и машинного обучения.
 37. Способы интеграция компонент ИИ в приложение.
 38. Паттерны проектирования для интеллектуальных систем
- CASE-средства для разработки интеллектуальных систем

6.4. Критерии оценивания

Посещаемость и выполнение заданий у доски - 0 до 15 баллов - максимум 7,5 % от общего числа баллов за семестр
Практические работы от 0 до 120 баллов - максимум - 60 % от общего числа баллов за семестр
Промежуточная аттестация (экзамен) максимум 10 баллов - 32,5 % от общего количества баллов

Если студент набирает:

- от 0 до 60 - оценка не удовлетворительно
- от 61 до 74 - оценка удовлетворительно
- от 75 до 84 - оценка хорошо
- от 85 до 100 - оценка отлично

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1. Рекомендуемая литература

7.1.1. Основная литература



	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л1.1	Новиков Ф. А.	Учебно-методическое пособие по дисциплине «Анализ и проектирование на UML» (http://e.lanbook.com/books/element.php?pl1_id=43540)	Санкт-Петербург : НИУ ИТМО, 2007	ЭБС
Л1.2	Косяков А., Свит У.	Системная инженерия. Принципы и практика (http://e.lanbook.com/books/element.php?pl1_id=66484)	Москва : ДМК Пресс, 2014	ЭБС
Л1.3	Джесугасан Р., Будро Д.	Реинжиниринг бизнеса: как грамотно внедрить автоматизацию и искусственный интеллект: практическое пособие (https://znanium.com/catalog/document?id=352155)	Москва : ООО "Альпина Паблишер", 2019	ЭБС
Л1.4	Бурков А.	Инженерия машинного обучения (https://e.lanbook.com/book/314834)	Москва : ДМК Пресс, 2022	ЭБС

7.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Ресурс
Л2.1	Буч Г., Рамбо Д., Якобсон И.	Язык UML. Руководство пользователя (http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=1246)	Москва : ДМК Пресс, 2008	ЭБС
Л2.2	Станкевич Л. А.	Интеллектуальные системы и технологии: учебник и практикум для вузов (https://urait.ru/bcode/530657)	Москва : Юрайт, 2023	ЭБС

7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Электроннобиблиотечная система издательства Лань https://e.lanbook.com
Э2	Электроннобиблиотечная система Юрайт https://urait.ru/
Э3	Электронно библиотечная система znanium https://znanium.com/

7.3 Перечень информационных технологий

7.3.1 Программное обеспечение

LMS Moodle
OpenOffice
Adobe Reader
NetBeans
Notepad++
Python
Visual Studio

7.3.2 Профессиональные базы данных и информационно-справочные системы

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Компьютеры, проектор
Доска маркеры (мел)

9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

--

10. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ ОБУЧАЮЩИМИСЯ С ИНВАЛИДНОСТЬЮ И ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием специальных технических средств и голо информационных технологий, предоставляемых Ресурсным учебно-методическим центром по обучению инвалидов и лиц с ограниченными возможностями здоровья ЧелГУ по запросу обучающегося.

1. Мобильные специальные технические средства для лиц с нарушениями зрения: портативный компьютер с вводом/выводом шрифтом Брайля с синтезатором речи «E1Braille-W14J G2»; ноутбуки с программной экранного доступа NVDA; электронные увеличители для удаленного просмотра; видеоувеличители портативные; тифлоплеер;



цифровые диктофоны.

2. Мобильные специальные технические средства для лиц с нарушениями слуха: система свободного звукового поля со встроенной совместимостью с FM-устройствами; радиоклассы «Сонет-PCM» с передатчиком, заушным индуктором и индукционной петлей; система информационная для слабослышащих переносная «Исток» А2 со встроенным плеером – звуковым информатором; документ-камера; программируемые слуховые аппараты индивидуального пользования.

3. Ассистивные информационные технологии: программное обеспечение экранного доступа с синтезом речи NVDA; программы экранного увеличения; программы речевого синтеза для компьютеров и ноутбуков; программы речевого синтеза для мобильных устройств; экранная клавиатура; экранная лупа.

При необходимости для обучающихся с нарушениями зрения на рабочих местах для проведения практических или лабораторных занятий устанавливается специальное программное обеспечение (программа речевой навигации NVDA, речевые синтезаторы, экранные лупы).

В учебные аудитории обеспечивается беспрепятственный доступ для обучающихся инвалидов и обучающихся с ограниченными возможностями здоровья. В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусматривается соответствующее количество мест для обучающихся с учетом нарушений их здоровья.

Для освоения дисциплины инвалидам и лицам с ограниченными возможностями здоровья предоставляется доступ к печатным источникам, имеющимся в научной библиотеке ЧелГУ, с помощью специальных технических средств; доступ к электронным источникам, представленным в форме электронного документа в фонде научной библиотеки ЧелГУ или электронно-библиотечных системах, с помощью специальных технических и программных средств (рабочее место для незрячего пользователя с программным обеспечением экранного доступа с синтезом речи NVDA, рабочее место с компьютерным роллером и клавиатурой Clevy с большими кнопками и с разделяющей клавиши накладкой).

Учебно-методические материалы для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме шрифтом Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

Для инвалидов и лиц с ограниченными возможностями здоровья освоение дисциплины может быть частично или полностью осуществлено с использованием дистанционных образовательных технологий (Moodle, Adobe Connect Pro и пр.).

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья используется индивидуальная работа. Под индивидуальной работой подразумевается две формы взаимодействия с преподавателем: индивидуальная учебная работа (консультации), т.е. дополнительное разъяснение учебного материала и углубленное изучение материала с теми обучающимися, которые в этом заинтересованы, и индивидуальная воспитательная работа. Индивидуальные консультации направлены на индивидуализацию обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или обучающимся с ограниченными возможностями здоровья.

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине обеспечивается выполнение следующих дополнительных требований в зависимости от индивидуальных особенностей, обучающихся:

- а) инструкция по порядку проведения процедуры оценивания предоставляется в доступной форме (устно, в письменной форме, в письменной форме шрифтом Брайля, устно с использованием услуг сурдопереводчика);
- б) доступная форма предоставления заданий оценочных средств (в печатной форме, в печатной форме увеличенным шрифтом, в печатной форме шрифтом Брайля, в форме электронного документа, задания зачитываются ассистентом, задания предоставляются с использованием сурдоперевода);
- в) доступная форма предоставления ответов на задания (письменно на бумаге, набор ответов на компьютере, письменно шрифтом Брайля, с использованием услуг ассистента, устно).

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Рабочая программа дисциплины "Анализ требований и проектирование систем искусственного интеллекта" по направлению подготовки (специальности) 01.03.02 "Прикладная математика и информатика" направленности (профилю) Прикладная математика и искусственный интеллект ФГБОУ ВО «ЧелГУ»

стр. 11

особенностями. Эти средства могут быть предоставлены ЧелГУ или могут использоваться собственные технические средства. При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на задания, процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Проведение процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья допускается с использованием дистанционных образовательных технологий.

Задание 1

Задание выполняется в группе из 3-4 человек

Использование системы управления требованиями (СУТ)

Выбирается одна из систем

<https://blog.themarfa.name/32-luchshiee-altiernativy-jira-dlia-upravlieniia-proiektami/>

Для описания требований выбирается одна из установленных программ на компьютере

Критерии оценивания

Выполнена регистрация участников группы в СУТ — 1 балл

Зафиксированы не менее 5 существующих требований к ПО (reverse engineering) — 5 баллов (по 1 баллу за каждое требование)

Предложено изменение требования, улучшение функциональности ПО - 2 балла, иначе 0 баллов

Выполнена классификация требований — 2 балла, иначе 0 баллов

Задание 2

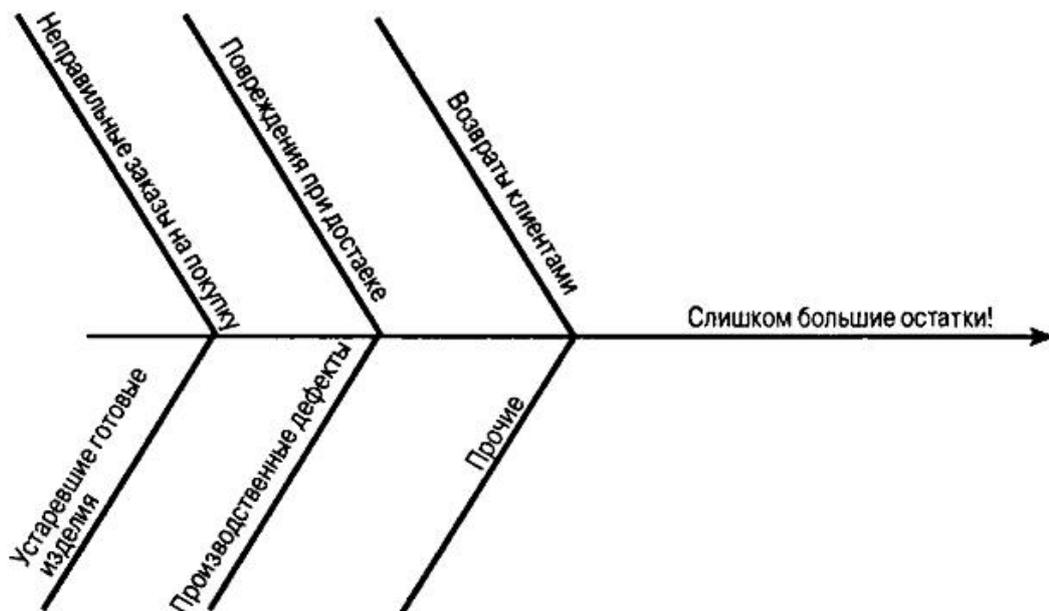
1. Поиск альтернатив и конкурентов
2. Выполнение этапов анализа проблемы

Этап 1. Достижение соглашения об определении проблемы

Простейший способ – просто записать проблему и выяснить, все ли согласны с такой постановкой. Примерная форма:

Элемент	Описание
Проблема	Описание проблемы
Воздействует на	Указание лиц, на которых оказывает влияние данная проблема
Результатом чего является	Описание воздействия данной проблемы на заинтересованных лиц и бизнес-деятельность
Выигрыш от	Указание предлагаемого решения
Может состоять в следующем	Список основных предоставляемых решением преимуществ

Этап 2. Выделение основных причин – проблем, стоящих за проблемой



Помогает методы анализа корневых причин (рыбий скелет). Многие корневые причины не стоят того, чтобы их устранять, поскольку затраты на их устранение превысят причиняемый проблемам ущерб. Поэтому необходимо определить вклад каждой корневой причины в проблему.

Критерии оценки

- Поставлена реальная проблема(ы) - 1 балл
- Указано не менее 5 причин для проблем - 5 баллов (по 1 баллу за причину)
- Есть причины причин - 1 балл
- Есть причины, которые можно решить программным путем - 1 балл
- Есть список из не менее 2 конкурирующих продуктов, которые будут использованы для сравнения (ссылки) - 2 балла

Максимальный балл — 10

Задание 3

Этап 3. Выявление заинтересованных лиц и пользователей

Заинтересованные лица – это прямые и не прямые пользователи, которых затрагивает реализация новой системы или приложения.

Непрямые пользователи - клиенты, надзорные организации, лица, осуществляющие сопровождение и эксплуатацию системы.

Этап 4. Определение границ системы-решения

Делим мир на две части – наша система и то, что взаимодействует с нашей системой.

Актант (актор) – это находящееся вне системы нечто (или некто), взаимодействующее с системой.

Этап 5. Выявление ограничений, налагаемых на решение

Ограничения уменьшают степень свободы, которой мы располагаем при предложении решения.

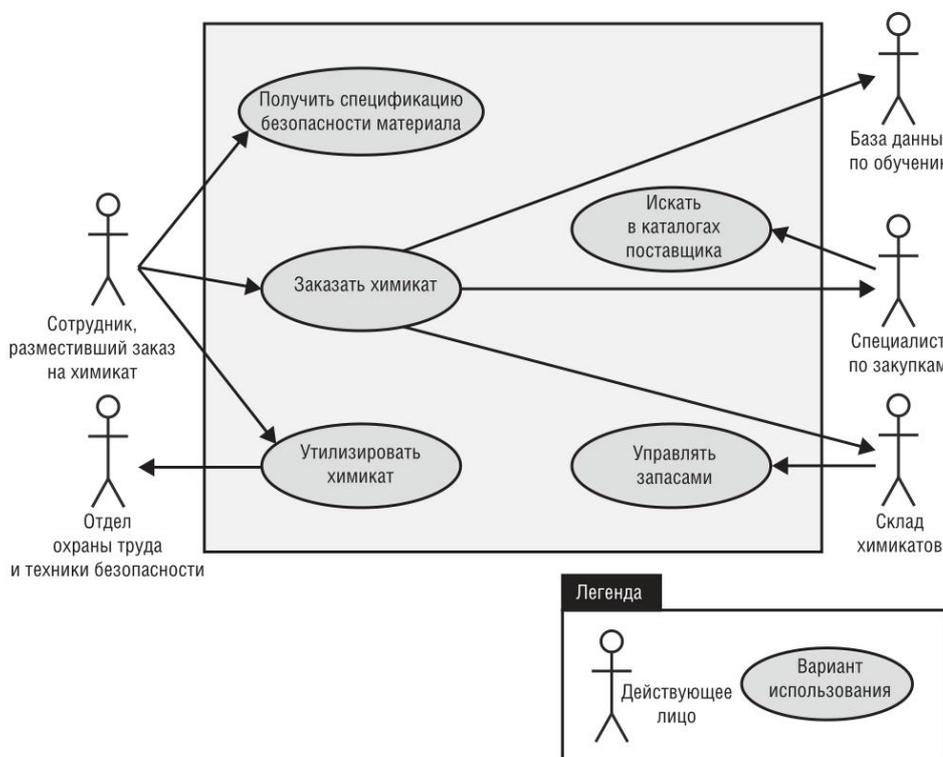
Некоторые из обнаруженных ограничений могут стать требованиями к системе.

Возможные источники: экономические, политические, технические, системные, эксплуатационные, на разработку.

Задание

Нарисовать диаграмму вариантов использования и дать описание одного из вариантов использования

Диаграмма вариантов использования



Примечание идентификатор используется для трассировки, чтобы не писать полное имя, идентификатор содержит тип и номер: UC (user case), ВИ (вариант использования). Номер может простым или составным (номер диаграммы.номер ВИ). Примеры UC-4, ВИ-2.5

Идентификатор и название: UC-4 Заказать химикат

Автор: Лори

Дата создания: 22.08.13

Основное действующее лицо: Сотрудник, разместивший заказ на химикат

Дополнительное действующее лицо: Поставщик; Склад химикатов; База данных по обучению

Описание: Сотрудник, разместивший заказ на химикат, указывает в запросе необходимый химикат, вводя его название или идентификатор или импортируя его структуру из соответствующего графического средства. Система выполняет запрос, предлагая контейнер с химикатом со склада или позволяя создать запрос на заказ у поставщика.

Триггер: Сотрудник указывает, что хочет заказать химикат.

Предварительные условия:

PRE-1. Личность пользователя аутентифицирована.

PRE-2. Пользователь имеет право запрашивать химикаты.

PRE-3. База данных по запасам химикатов в данный момент доступна.

Выходные условия:

POST-1. Запрос сохраняется в Chemical Tracking System.

POST-2. Запрос отправлен на склад химикатов или поставщику.

Нормальное направление развития варианта использования:

4.0 Запросить химикат со склада

1. Сотрудник указывает требуемый химикат.
2. Система перечисляет контейнеры с необходимым химикатом, имеющиеся на складе.
3. Сотрудник может просмотреть историю любого контейнера.
4. Сотрудник выбирает определенный контейнер или просит отправить запрос поставщику (см. 4.1).
5. Сотрудник вводит остальную информацию, чтобы завершить запрос.
6. Система сохраняет запрос и отправляет его на склад химикатов.

Альтернативное направление развития варианта использования:

4.1 Запросить химикат у поставщика

1. Сотрудник ищет химикат по каталогам поставщика (см. 4.1.E1).
2. Система отображает список поставщиков, где также указаны размеры, класс и цена контейнеров.
3. Сотрудник выбирает поставщика, размер, класс и количество контейнеров.
4. Сотрудник вводит остальную информацию, необходимую для запроса.
5. Система сохраняет запрос и перенаправляет его поставщику.

Исключения:

4.1.E1. Химиката нет в продаже

1. Система отображает сообщение «У поставщиков нет такого химиката».
2. Система предлагает сотруднику запросить другой химикат или выйти из программы.

3а. Сотрудник просит запросить другой химикат.

4а. Система заново начинает нормальное направление варианта использования.

3б. Сотрудник решает выйти из системы.

4б. Система завершает вариант использования.

Приоритет: Высокий

Частота использования: Примерно пять раз в неделю каждым химиком, 200 раз в неделю каждым работником склада.

Бизнес-правила: BR-28, BR-31

Специальные требования: Система должна импортировать химические структуры в стандартной закодированной форме из любых средств, поддерживающих рисование химических структур.

Предположения: Импортированные химические структуры должны быть верными.

Критерии оценки

- Нарисована диаграмма вариантов использования для всех прямых пользователей - 3 балла
- Варианты использования соответствуют потребностям и целям пользователей - 2 балла
- Описан один из вариантов использования - 3 балла
- В описании нет противоречий - 2 балла

Максимальный балл — 10

Задание 4

Выбор и оценка задач для автоматизации

Критерии оценки

- Выявлены задачи для автоматизации — 3 балла (1 балл за каждую задачу, но не более 3 баллов)
- Определена пригодность методов искусственного интеллекта и машинного обучения- для каждой задачи 3 балла (оценка снижается на 1 балл за каждую ошибку в оценке)
- Выполнена классификация всех задач для автоматизации - 3 балла (оценка снижается на 1 балл за каждую ошибку)
- Выполнена оценка окупаемости (целесообразности) разработки - 1 балл

Максимальный балл — 10

Задание 5

Оценивание пригодности исходных данных, эффекта от использования результатов

Вопросы к данным

- Доступны ли данные?
- Насколько велик объем данных?
- Пригодны ли данные для использования?
- Понятны ли данные?
- Надежны ли данные?
- Есть ли проблемы с данными? (Высокая стоимость, Плохое качество? Зашумленность, Смещение, Низкая предсказательная способность, Устаревшие примеры, Выбросы, Просачивание данных)
- Нужна ли обработка?
- Где хранятся данные?

Критерии оценки

- Определен набор исходных данных для решения задачи — 3 балла (оценка снижается на 1 балл за каждую ошибку)
- Выполнена оценка исходных данных по списку вопросов — 4 балла (оценка снижается на 1 балл за каждую ошибку)
- Выполнена оценка эффекта от использования результатов - 3 балла (оценка снижается на 1 балл за каждую ошибку)

Задание 6

Задание выполняется в группе из 3-4 человек, раздел 5 выполняется совместно, остальные разделы распределяются между участниками группы.

Написать документ-концепцию

Документ-концепция описывает приложение в общих чертах, а также содержит описания целевых рынков, пользователей системы и функций приложения (как предлагаемых к реализации в версии 1, так и выявленные), ограничения системы.

Содержание документа-концепции

1. Введение

В данном разделе необходимо представить общую характеристику документа-концепции в целом

1.1. Цель документа-концепции

Цель данного документа состоит в сборе, анализе и определении высокоуровневых потребностей пользователей и функций продукта.

1.2. Общая характеристика продукта

В данном разделе определяется цель приложения, его версия и новые предоставляемые функции. Здесь следует

- указать продукт или приложение, которое создается или изменяется;
- дать общее описание того, что продукт будет делать и, если необходимо, чего не будет делать;
- описать применение продукта, в том числе достижимые с его помощью выгоды, цели и задачи.

1.3. Ссылки

Этот подраздел содержит список всех документов, упоминаемых где-либо в документе-концепции и список источников, к которым можно обратиться за справками.

2. Описание пользователя

2.1. Характеристика рынка/пользователя

Здесь необходимо кратко перечислить основные характеристики рынка, которые послужили мотивацией решений, касающихся продукта: описать и указать целевые сегменты, а также оценить объем и перспективы роста рынка, ориентируясь на число потенциальных пользователей или количество денег, которые в настоящее время тратят ваши заказчики при решении задач.

2.2. Описания пользователей

Здесь следует описать все типы пользователей. Для каждого типа пользователей указать

- Технический уровень и опыт
- Основные обязанности

- Тенденции, упрощающие или усложняющие работу пользователя
- В чем пользователь видит успех и как пользователь вознаграждается

2.3. Среда пользователя

- Сколько человек участвует в выполнении данной задачи? Изменится ли их число?
- Сколько времени длится цикл выполнения задачи? Изменится ли это?
- Существуют ли некие уникальные ограничения среды: на мобильную связь, по работе вне помещения, в полете и т.д.
- Какие системные платформы используются в настоящее время? Какие платформы предполагается использовать в будущем?
- Какие еще приложения используются? Должно ли ваше приложение объединяться с ними?

2.4. Основные потребности пользователя

Следует перечислить основные проблемы или потребности так, как они осознаются пользователем. Для каждой проблемы нужно прояснить следующие моменты

- В чем причины данной проблемы?
- Как она решается в настоящее время?
- Какие решения представляет себе пользователь?

2.5. Альтернативы и конкуренты

Нужно указать возможные альтернативы поведения пользователя. Среди них может быть покупка продукта конкурентов, создание собственного решения или просто сохранение существующей ситуации. Опишите основные преимущества и недостатки каждого варианта с точки зрения конечного пользователя.

2.5.1. Конкурент 1

3. Характеристика продукта

В данном разделе предлагается общее описание возможностей продукта, интерфейсов с другими приложениями и конфигураций систем.

3.1. Общее описание продукта

В данном подразделе следует описать, как продукт взаимодействует с другими связанными с ним продуктами и средой пользователя. Если продукт является независимым и самодостаточным, это необходимо указать. Если продукт является компонентом более крупной системы, в данном подразделе необходимо описать, как эти системы взаимодействуют, а также указать соответствующие интерфейсы между системами.

3.2. Определение позиции продукта

Предлагается общее определение, характеризующее на самом высоком уровне абстракции назначение продукта и важность проекта.

3.3. Краткий обзор возможностей

Краткая характеристика основных возможностей и функций продукта с точки зрения преимуществ.

3.4. Предположения и зависимости

Описываются предположения, изменение которых приведет к изменению концепции

продукта.

3.5. Вопросы затрат и цены

4. Атрибуты функций

Функции имеют атрибуты, предоставляющие дополнительную информацию, которую можно использовать для оценки, отслеживания и определения очередности предлагаемых для реализации элементов разработки, а также управления ими. Нужно описывать в данном разделе только те атрибуты (и их значения), которые вы выберете, чтобы все участники могли лучше понять содержание каждой функции.

4.1. Статус (предложена, принята, включена) (более подробно в файле Атрибуты.docx)

4.2. Приоритет (критический, важный, полезный)

4.3. Уровень трудозатрат

4.4. Риск

4.5. Стабильность

4.6. Целевая версия

4.7. Кому предназначена

4.8. Обоснование

5. Функции продукта

Поскольку документ-концепция изучается широким кругом причастных к проекту лиц и служит основой для достижения соглашения, функции должны описываться на естественном языке пользователя. Описание функции должно быть кратким и ясным, как правило, одно-два предложения. Для эффективного управления сложностью приложения мы рекомендуем, чтобы описание возможностей любой новой системы (или усовершенствования существующей) производилось на достаточно высоком уровне абстракции и состояло из 25-99 функций. Эти функции составляют основу для определения продукта, а также управления масштабом и проектом в целом. Каждая из них будет описана более подробно в последующих спецификациях.

5.1. Функция 1

5.2. Функция 2

6. Основные варианты использования

Следует описать несколько основных вариантов использования, которые важны для архитектуры или лучше всего помогут читателю понять, как предполагается использовать систему

7. Другие требования к продукту (ограничения)

7.1. Применяемые стандарты

7.2. Системные требования

7.3. Лицензирование и инсталляция

7.4. Требования производительности

8. Требования к документации

В данном разделе описывается, какую документацию необходимо разработать для поддержки успешного внедрения приложения.

8.1. Руководство пользователя

8.2. Интерактивная подсказка

8.3. Руководства по инсталляции, конфигурация и файл ReadMe

8.4. Маркировка и упаковка

9. Глоссарий

Глоссарий описывает все присущие данному проекту термины, в том числе все аббревиатуры, которые могут быть непонятны пользователю или другим читателям данного документа.

Критерии оценки

- Структура документа-концепции соответствует шаблону - 2 балла
- Достаточное описание пользователя и его потребностей (пп 2.2,2.3,2.4) - 3 балла
- Рассмотрено не менее 2 альтернатив - 2 балла
- В п.3 указано достаточное обоснование важности проекта - 1 балл
- Указано не менее 5 функций - 2 балла

Итого 10 баллов максимум

4. Атрибуты функций

Как и требования, функции имеют атрибуты, предоставляющие дополнительную информацию, которую можно использовать для оценки, отслеживания и определения очередности предлагаемых для реализации элементов разработки, а также управления ими. Ниже мы описали атрибуты, которые можно использовать в документе-концепции. *Вам нужно описывать в данном разделе только те атрибуты (и их значения), которые вы выберете, чтобы все участники могли лучше понять содержание каждой функции.*

4.1. Статус

Задается в результате переговоров и рассмотрения руководством проекта. Информация о статусе отражает ход процесса определения базового уровня проекта. Атрибут статуса функции может иметь следующие значения.

Предложена. Используется для описания обсуждаемых функций, которые еще не рассмотрены и не приняты "официальным органом" — рабочей группой, состоящей из представителей команды проекта* руководства и пользователей или заказчиков.

Принята. Возможности, которые "официальный орган" признал полезными и достижимыми и принял к реализации.

Включена. Функции, включенные в базовый уровень на данный момент времени.

4.2. Приоритет

Приоритеты функций продукта задаются представителями маркетинга, менеджером продукта или аналитиком базового уровня. Упорядочение функций по их относительной важности для конечного потребителя открывает диалог между заказчиками, аналитиками и членами команды разработчиков. Приоритеты используются для управления масштабом и определения очередности разработки. Ниже предложена одна из возможных схем задания приоритетов.

Критический. Основные функции. Если их не удастся реализовать, система не будет удовлетворять потребности заказчика. В версии должны быть реализованы все критические функции, в противном случае график является нереальным.

Важный. Функции, важные для успешной и эффективной работы системы в большинстве приложений. Данные функциональные возможности нельзя легко обеспечить иным способом. Если важные функции не войдут в реализацию, это может повлиять на удовлетворение пользователя или заказчика результатом работы или даже на доходы от продаж, но выпуск версии не должен задерживаться из-за нехватки некой важной функции.

Полезный. Функции, которые нужны в менее распространенных приложениях, будут использоваться не так часто или их можно достаточно эффективно заменить другими действиями. Если они не войдут в реализацию, это не окажет заметного воздействия на отношение заказчика или доходы.

4.3. Уровень трудозатрат

Определяется командой разработчиков и используется для управления масштабом и определения очередности разработки. Поскольку некоторые функции требуют больше времени и ресурсов, чем другие, оценка количества командо- или человеко-недель, строк кода или функциональных единиц помогает соразмерить сложность и оценить, что можно, а что нельзя осуществить за определенный период времени.

4.4. Риск

Задается командой разработчиков на основе вероятности того, что данная функция вызовет нежелательные последствия для проекта, такие как превышение средств, отставание от графика или даже закрытие проекта. Большинство менеджеров продукта считают достаточным деление рисков на категории *низкий, средний, высокий*, хотя возможна и более тонкая градация. Иногда риск можно оценить, измеряя меру неопределенности (диапазон) оценок времени работы команды.

4.5. Стабильность

Определяется аналитиком и командой разработчиков, исходя из вероятности того, что может измениться данная функция или понимание командой этой функции. Эта информация используется для того, чтобы помочь при определении приоритетов разработки и выявить те элементы, для которых следующим действием должно стать дополнительное исследование.

4.6. Целевая версия

Записывается, в какой версии продукта предполагается впервые реализовать данную функцию. Это поле можно использовать, чтобы поместить функции в базовый уровень конкретной версии. Комбинируя этот атрибут с полем статуса, команда может предлагать, записывать и обсуждать для версии различные функции, не приступая к их разработке. Будут реализовываться только функции, имеющие статус "Включенная", для которых определена целевая версия. При необходимости сокращения масштаба номер целевой версии может быть увеличен, так что элемент остается в документе-концепции, но его реализация будет отложена на более поздний срок.

4.7. Кому предназначена

Во многих проектах функции будут предназначаться "функциональным группам", ответственным за их дальнейшее исследование, написание программных требований, а также, возможно, реализацию. Это помогает членам команды разработчиков лучше понять свои обязанности.

4.8. Обоснование

Данное текстовое поле используется для отслеживания источника запрашиваемой функции. В этом поле записывается объяснение причины существования данной функции или ссылка на него. Например, ссылка может указывать на страницу, номер строки спецификации требований к продукту или временной маркер на видеозаписи важного интервью с клиентом.

Информационная система «Электронный архив»

Документ-концепция

© Е. П. Моргунов

История исправлений и дополнений

Дата	Версия	Описание	Автор
01.04.2005	0.5	Исходная версия	Е. П. Моргунов
20.06.2009	1.0	Проведены реструктуризация и дополнение документа с учетом рекомендаций, предложенных в книгах И. Соммервилла и Д. Леффингуэлла (их библиографические описания см. ниже)	Е. П. Моргунов

Примечание. Дата выпуска версии 0.5 – приблизительная.

1. Введение

1.1. Цель документа-концепции

Цель данного документа состоит в сборе и анализе исходной информации для разработки, определении высокоуровневых потребностей пользователей и формулировании функций продукта.

1.2. Назначение и общая характеристика продукта

Настоящий продукт предназначен для решения разнообразных задач управления информационным массивом, который накапливается и используется любым человеком в процессе интеллектуальной деятельности (в том числе, и связанной с отдыхом). Информация, интересующая конкретного человека, может содержаться в печатных и электронных изданиях, аудио- и видеоматериалах, рукописях, ксерокопиях и т. д. Зачастую требуется не только отыскать нужную информацию, например, цитату, но также и указать ее источник. Следовательно, возникает необходимость накопления и структуризации как содержательной информации, ради получения которой и проводится проработка различных источников (их чтение, просмотр, выписывание цитат и т. п.), так и библиографической информации. Ее можно условно назвать метаинформацией.

Программный продукт в технологическом плане будет представлять собой информационную систему, построенную на основе «большой» СУБД и Web-технологий. Это позволит в качестве клиентского места использовать Web-браузер, а при необходимости – организовать удаленный доступ к базе данных через Internet/Intranet. В содержательном плане программный продукт должен выполнять следующие функции:

– электронный каталог библиографических записей о книгах, журналах, статьях и др., имеющихся у пользователя или заинтересовавших его, т. е. попавших в его поле зрения, в сферу его интересов;

– хранилище цитат из проработанных источников;

– система учета экземпляров книг, журналов и других объектов, хранящихся у пользователя, а также, если это необходимо пользователю, в библиотеках или у других лиц;

– механизм связывания библиографических записей с полнотекстовыми документами в электронной форме (при их наличии), позволяющий организовать быстрый доступ к таким документам.

Использование программного продукта позволит:

– экономить время, традиционно затрачиваемое на поиск библиографической информации и цитат в массиве неупорядоченных файлов и печатных материалов;

– экономить время, требующееся на оформление библиографических списков в рефератах, курсовых работах, научных статьях, диссертациях. Удобнее получать всю информацию библиографического характера из единого центра, только один раз затратив время на корректное оформление каждой записи, вновь вводимой в базу данных электронного архива;

– организовать обмен библиографической информацией с другими студентами, аспирантами, коллегами;

– повысить уровень культуры в работе с библиографической информацией.

Сферы применения программного продукта:

– индивидуальное использование;

– коллективное использование, например, в научном коллективе, занимающемся работами по одной тематике.

1.3. Ссылки и использованная литература

1.3.1. Список документов, упоминаемых в документе-концепции

1. Архитектура и системные требования (спецификация).

1.3.2. Список источников, к которым можно обратиться за справками в процессе разработки

1. Стандарты:

– ГОСТ 7.1–2003 «СИБИД. Библиографическая запись. Библиографическое описание. Общие требования и правила составления»;

– ГОСТ 7.11–2004 «СИБИД. Библиографическая запись. Сокращение слов и словосочетаний на иностранных европейских языках»;

– ГОСТ 7.12–93 «СИБИД. Библиографическая запись. Сокращение слов на русском языке. Общие требования и правила»;

– ГОСТ 7.60–90 «СИБИД. Издания. Основные виды. Термины и определения»;

– ГОСТ 7.80–2000 «СИБИД. Библиографическая запись. Заголовок. Общие требования и правила составления»;

– ГОСТ 7.82–2001 «СИБИД. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления»;

– ОСТ 29.130-97 «Издания. Термины и определения».

2. Система автоматизации библиотек ИРБИС. Общее описание системы [Текст]. – М. : ГПНТБ России, 2002. – 260 с.

2. Описание пользователей

2.1. Виды пользователей и их краткие описания

2.1.1. Ученый

Имеет высокий уровень культуры интеллектуального труда. Читает много разнообразной научной литературы, как периодической, так и непериодической. Пишет научные статьи, доклады для выступления на конференциях и учебно-методические работы.

2.1.2. Аспирант

Стремится стать ученым и старается делать то же, что описано в п. 2.1.1.

2.1.3. Студент

Объем прорабатываемой научной и учебной литературы – меньше, чем у ученого или аспиранта. Однако объем прочитываемой художественной, научно-популярной и развлекательной литературы может быть большим, чем у ученого, т. к. студент имеет больше свободного времени. Кроме того, студент может иметь в качестве хобби, например, коллекционирование аудиозаписей.

2.1.4. «Обычный» человек

Среди людей, не относящихся к категории ученых, также нередко встречаются любители книг, имеющие большие домашние библиотеки. Многие люди накапливают архивы фото-, аудио- и видеоматериалов, которые трудно содержать в порядке без использования компьютерной системы учета.

2.2. Среда пользователя

Большинство пользователей используют операционную систему (ОС) Windows, хотя в настоящее время все большую популярность набирает ОС Linux, относящаяся к классу UNIX-подобных ОС.

Зачастую проработанные литературные источники нигде не фиксируются, выписки, сделанные из них, находятся в разрозненных файлах или на бумажных носителях. Для формирования списков использованных источников (библиографических списков), например, при написании реферата или статьи каждый раз используется метод «с нуля». Поэтому пользователям трудно выполнить библиографические описания в соответствии с ГОСТами

2.3. Основные потребности пользователя

Общая потребность (возможно, не до конца осознаваемая пользователями в силу разных причин): получить инструмент, позволяющий упорядочить всю библиографическую информацию, которой вынужден оперировать пользователь, и, тем самым, сократить затраты времени на эту деятельность. Также важной потребностью может являться создание электронного каталога домашней библиотеки (включающей не только книги, но также и видео-, фото-, аудиоматериалы).

2.3.1. Ученый

Необходима возможность отслеживания выхода новых выпусков периодических изданий и их оперативной проработки, сохранения цитат в базе данных, формирования списков литературы при написании научных статей и учебно-методических работ.

2.3.2. Аспирант

То же, что и для ученого.

2.3.3. Студент

То же, что и для ученого, с той лишь разницей, что студент не пишет учебно-методических работ, но пишет рефераты, курсовые и дипломные работы.

2.3.4. «Обычный» человек

Необходим инструмент, который мог бы помочь в упорядочивании домашней библиотеки, фонотеки, видеотеки.

3. Состояние рынка и конкурирующие продукты

3.1. Характеристика рынка

На исследуемом рынке представлены два типа программных продуктов:

– «большие» профессиональные автоматизированные библиотечные информационные системы (АБИС), такие, как «ИРБИС», «Академия+» и др., предназначенные для использования в библиотеках;

– «малые» информационные системы, предназначенные для домашнего использования. Последние, как правило, являются программами-каталогизаторами, т. е. позволяют вводить в базу данных описания книг и выполнять выборки из базы данных по различным критериям.

«Большие» АБИС не подходят для персонального использования, т. к. они чрезмерно сложны для пользователя, не имеющего специального библиографического образования. Кроме того, они содержат целый ряд подсистем, предназначенных для выполнения функций, совершенно не нужных в домашних условиях (например, определение книгообеспеченности, списание книг, учет читателей и т. д.). Немаловажным фактором является и тот факт, что подобные системы не являются свободно-распространяемыми – они стоят дорого.

«Малые» же информационные системы, напротив, слишком просты. Они, как правило, хорошо выглядят с технологической стороны: имеют возможности настройки интерфейса пользователя, умеют добывать описания книг на Web-сайтах книжных магазинов (например, Amazon.com), позволяют настроить шрифты, добавить новые таблицы в базу данных и новые поля в таблицы и т. д. Однако с точки зрения соответствия стандартам, принятым в библиотечном деле, эти продукты далеки от совершенства.

Таким образом, необходим программный продукт, условно говоря, среднего класса, который сочетал бы в себе сильные стороны программных продуктов обоих типов, но при этом не был бы слишком сложным в освоении его пользователем, не имеющим специальных знаний.

Потенциальный круг пользователей весьма широк: от ученых до аспирантов и студентов, от работников умственного труда до обычных любителей книги, коллекционирующих также видеофильмы и музыкальные произведения.

Основные технологии, применяемые в отрасли для разработки подобных программных продуктов, следующие: языки программирования – Delphi, C/C++, Visual Basic; СУБД – CDS/ISIS (ЮНЕСКО), Microsoft Access и др. «Малые» информационные системы не работают под управлением операционной системы UNIX (Linux, FreeBSD).

3.2. Конкурирующие программные продукты

3.2.1. Общая характеристика

К недостаткам существующих «больших» библиотечных информационных систем (при оценке их с позиции индивидуального использования в домашних условиях) можно отнести следующее:

- большое количество функций, не требующихся обычному пользователю;
- сложная система ввода данных, требующая от пользователя специальных знаний в области библиографической деятельности;
- высокая цена программного продукта;
- отсутствие возможности сделать собственные пометки к проработанным источникам (книгам, статьям и т. д.) и сохранить цитаты из этих источников.

Особенностью «больших» библиотечных информационных систем является то, что библиографическое описание в них формируется алгоритмическим способом в соответствии с библиографическими стандартами из элементарных данных, вводимых пользователем.

К недостаткам существующих «малых» информационных систем можно отнести следующее:

- отсутствие возможности формирования библиографических описаний, соответствующих стандартам (ГОСТ 7.1–2003 или аналогичному международному стандарту). Вместо цельного библиографического описания имеется только набор сведений об объектах, внесенных в базу данных: авторы, заглавие, место издания и т. д. В отчетах эти сведения группируются в некое подобие описания, но оно не соответствует никаким стандартам;
- отсутствие развитой возможности конфигурирования программного продукта (с использованием конфигурационного файла или базы данных). Конфигурирование сводится, в основном, к настройке внешнего представления программы и данных, но оно мало помогает в ускорении работы оператора при вводе данных;
- неразвитость системы вспомогательных средств, сокращающих объем работы пользователя при вводе однотипных данных (подобные средства есть во всех Internet-браузерах);

3.2.2. Система автоматизации библиотек «ИРБИС»

Разработчик: Государственная публичная научно-техническая библиотека России (ГПНТБ). Сайты: <http://www.gpntb.ru>, <http://www.elnit.org>.

«Большая» библиотечная информационная система. Очень мощная, сложная, дорогая.

Фрагмент описания программного продукта, взятый с сайта разработчика (стиль и орфография сохранены)

Основные характеристики:

- поддержка произвольного количества баз данных, составляющих Электронный каталог или представляющих собой проблемно-ориентированные библиографические базы данных;
- технология автоматического формирования словарей, на основе которых реализуется быстрый поиск по любым элементам описания и их сочетаниям;
- средства для ведения и использования Авторитетных файлов, баз данных УДК, ББК, ГРНТИ и Тезауруса;
- поддержка традиционных «бумажных» технологий: от печати форм заказа/подписки и листов книги суммарного учета до печати всех видов каталожных карточек;
- технологии, ориентированные на использование штрих-кодов и радиометок на экземплярах изданий и читательских билетах;
- поддержка многоязычия на основе UNICODE, т. е. возможность ввода на любых языках мира;
- поддержка ссылок от библиографических описаний на полные тексты, графические данные и другие внешние объекты (включая ресурсы Интернет);
- средства для создания и ведения полнотекстовых баз данных (электронной библиотеки);
- специальные средства для создания имидж-каталогов по ретрофонду библиотеки на основе графических образов каталожных карточек и автоматического распознавания их текстов;
- средства для перевода пользовательских интерфейсов на другие языки;
- широкий набор сервисных средств, обеспечивающих удобство и наглядность пользовательских интерфейсов, упрощающих процесс ввода, исключая ошибки и дублирование информации;
- широкие возможности для адаптации к условиям работы конкретной библиотеки, включая средства создания уникальных рабочих профилей для всех категорий пользователей;
- открытость, позволяющая пользователю самостоятельно вносить изменения в широких пределах: от изменения входных и выходных форм до разработки оригинальных приложений.

«Оригинальное программное обеспечение системы написано на Delphi с использованием библиотеки ISIS32.DLL (Vireme, Бразилия). Физическая структура БД соответствует СУБД CDS/ISIS (ЮНЕСКО)» (цитата из описания программного продукта).

3.2.3. Автоматизированная библиотечная информационная система «Академия+»

Разработчик: Центр автоматизированных технологий «Ростехноком». Сайты: <http://www.rostechnocom.ru>; <http://www.academy-plus.ru>.

«Большая» библиотечная информационная система. Очень мощная, сложная, дорогая.

Фрагмент описания программного продукта, взятый с сайта разработчика (стиль и орфография сохранены)

Основные характеристики:

- масштабируемая трехуровневая клиент-серверная архитектура;
- независимость от программно-аппаратной платформы (любая аппаратная платформа: IBM PC, SUN и др.; любая операционная система: Windows, UNIX, LINUX);
- независимость от СУБД (любая реляционная СУБД: ORACLE, MS SQL Server, My SQL, PostgreSQL и др.);
- работа системы в Internet и Intranet без ограничения количества пользователей;
- поддержка UNICODE и штрих-кодирования на программном уровне.

3.2.4. BookCAT

Разработчик: FNProgramvare (Норвегия). Сайт: <http://www.fnprg.com>.
«Малая» информационная система. Самая развитая из систем подобного класса.

3.2.5. Учет книг

Разработчик: «Простой Софт». Сайт: <http://www.simple-soft.ru>,
<http://www.prostoysoft.ru>.

Фрагмент описания программного продукта, взятый с сайта разработчика (стиль и орфография сохранены)

Основные функции программы

Ведение базы книг, журналов. Каталогизация

В базе данных содержится информация о книгах, журналах. Предусмотрены такие поля как – название, авторы, категория, тип, издательство, серия, формат, год издания, количество страниц, тираж, обложка, ISBN, УДК, № шкафа, № полки, блок, подблок, время добавления и т. д. Для каждой книги показываются все ее читатели (которые читали эту книгу ранее и читают сейчас).

Предусмотрены удобные способы сортировки и фильтрации данных, что позволяет быстро найти нужные книги. Любую таблицу базы можно распечатать, экспортировать в MS Word, MS Excel или текстовый формат CSV. Имеется импорт из других источников данных в формате CSV.

Учет должников по возврату книг, журналов

Система фиксирует информацию о читателях – ФИО, контактная информация, выданные книги, даты выдачи и возврата книг. Контролируя значение поля с датой возврата книги можно легко вести учет должников. Таблица «На руках» показывает список всех выданных на руки книг и журналов.

Функциональные возможности программы

С помощью программы вы сможете делать следующее:

Создавать, изменять, удалять записи, поля, таблицы.

Импортировать данные в любую таблицу базы данных из текстовых файлов.

Удалять дублированные записи с одинаковым названием и автором. Можно настроить по-другому.

Сортировать таблицы по любому полю, включая сортировку по нескольким полям (до 3-х) удерживая клавишу Shift.

Фильтровать таблицу по любому полю, используя следующие операторы: =, >, >=, <, <=, <>, «Содержит», «Не содержит», «Начинается с», «Не начинается с», «Кончается на», «Не кончается на», LIKE, NOT LIKE.

Группировать одинаковые данные в любом поле, когда таблица отсортирована по этому полю (для отмеченных полей в свойствах таблицы).

Помечать записи как «Избранное», тогда они будут отображаться оранжевым цветом. Цвет задается в свойствах таблицы.

Помечать записи как «Мертвое» («Неинтересное»), тогда они будут отображаться серым (или другим) цветом.

Настраивать правила цветовыделения. Вы сами определяете, какие строки, каким цветом и при каких условиях выделять.

Строить дерево по любым полям с произвольным количеством уровней для иерархического отображения данных любой таблицы.

Изменять данные в любом поле (кроме ID и вычисляемых полей) прямо в таблице или в отдельной форме (выбирается в настройках), отмечать несколько записей, удалять, печатать, экспортировать отмеченные.

Изменять или удалять сразу несколько записей в любой таблице базы данных с помощью формы «Групповое обновление».

Создавать новые хранимые поля для таблиц следующих типов: текстовое, числовое, Да/Нет, Дата и время.

Создавать вычисляемые поля для таблиц, например можно создать поле с формулой «[Поле 1] / [Поле 2]».

Создавать вычисляемые поля, значения которых будут браться из других таблиц. Например, можно вывести имя должника из таблицы «Должники».

Создавать новые таблицы с абсолютно такими же возможностями по действиям с ними, как и у любой другой таблицы

Привязывать ниспадающие списки полей к другим таблицам для легкого выбора значений из них при редактировании в таблице или для выбора из других форм при редактировании в форме.

Задавать произвольное количество подчиненных таблиц для любой таблицы, для чего необходимо задать привязку по полям в свойствах таблицы.

Менять порядок следования полей в любой таблице, используя перетаскивание или с помощью формы «Настройки».

Переименовывать поля таблиц и названия самих таблиц в соответствии со спецификой вашего бизнеса. (Точнее говоря, их лейблы.)

Печатать текущее представление любой таблицы с учетом видимости полей, их ширины и порядка.

Экспортировать данные любой таблицы в MS Excel или текстовый CSV-файл с учетом текущего представления таблицы.

Экспортировать текущую запись в MS Word на основе файла-шаблона с закладками, соответствующими названиям полей.

Работать с несколькими файлами баз данных, создавать новые базы данных, разумеется, можно также открывать их с помощью MS Access.

3.2.6. eLibPro

Разработчик: Songs Technologies. Сайт: <http://songstech.com>.

«Малая» информационная система. Использует формат данных mdb (Microsoft Access).

3.2.7. Librarian Pro

Разработчик: Koingo Software. Сайт: <http://www.koingosw.com>.

«Малая» информационная система.

3.2.8. HomiStorage

Разработчик: Hominoid Software. Сайт: <http://www.hominoid.nm.ru>.

«Малая» информационная система.

3.2.9. All My Books

Разработчик: Bolide Software. Сайт: <http://www.bolidesoft.com>.
«Малая» информационная система.

3.3. Определение позиции продукта на рынке

Для ученых, аспирантов, студентов и всех других пользователей, которые хотят иметь возможность упорядочить свою коллекцию книг, журналов, видео- и аудиоматериалов, рукописей и электронных документов. «Электронный архив» является автоматизированной информационной системой для управления домашней библиотекой.

Наш продукт позволит всегда иметь под рукой качественную библиографическую информацию.

В отличие от конкурирующих «малых» информационных систем, таких, как BookCAT, «Учет книг» и др., наш продукт разработан с учетом требований стандартов на библиографическое описание, использует «большую» СУБД, имеет развитые средства помощи пользователю для ускорения ввода данных.

4. Атрибуты функций

4.1. Приоритет

Критический. Основные функции. Если их не удастся реализовать, система не будет удовлетворять потребности заказчика. В версии должны быть реализованы все критические функции, в противном случае график является нереальным.

Важный. Функции, важные для успешной и эффективной работы системы в большинстве приложений. Данные функциональные возможности нельзя легко обеспечить иным способом. Если важные функции не войдут в реализацию, это может повлиять на удовлетворение пользователя или заказчика результатом работы или даже на доходы от продаж, но выпуск версии не должен задерживаться из-за нехватки некой важной функции.

Полезный. Функции, которые нужны в менее распространенных приложениях, будут использоваться не так часто или их можно достаточно эффективно заменить другими действиями. Если они не войдут в реализацию, это не окажет заметного воздействия на отношение заказчика или доходы.

4.2. Уровень трудозатрат

Низкий. Требуется не более 1 человеко-недели на разработку.

Средний. Требуется от 1 до 4 человеко-недель на разработку.

Высокий. Требуется более 4 человеко-недель на разработку.

4.3. Риск

Указывается симптомы, по которым функция может быть отнесена к соответствующему уровню риска и возможные последствия при реализации риска.

Низкий. Назначение хорошо понято. Ресурсы требуемой квалификации доступны в полном объеме. Разрабатываемые подсистемы не потребуют новой технологической платформы. Превышение сроков и бюджета при реализации функции маловероятно.

Средний. Назначение и рамки определены более-менее четко. Ресурсы требуемой квалификации доступны в основном. Подсистемы создаются на новой, но стабильной технологической платформе. Последствия — функция может быть реализована с превышением бюджета или сроков не более 1,5-2 раз.

Выше среднего. Назначение и рамки определены недостаточно четко. Ресурсы требуемой квалификации сильно ограничены. Подсистемы создаются на новой технологической платформе, сомнения в рыночной стабильности платформы. Последствия — функция может быть реализована с превышением бюджета или сроков в 2 или более раз.

Высокий. Назначение и рамки определены нечетко. Ресурсы требуемой квалификации практически отсутствуют. Подсистемы создаются на новой технологической платформе, в отношении которой крайне мало ясности. Последствия — функция может быть не реализована.

4.4. Целевая версия

Записывается, в какой версии продукта предполагается впервые реализовать данную функцию.

5. Функциональные требования (функции продукта)

5.1. Обязательные функции для первой версии

Название «первая» является условным. Номер версии должен быть определен на основе фактического уровня готовности программного продукта. Предположительно это будет версия 0.5.

ПФТ1. Соответствие требованиям ГОСТ 7.1–2003

1. Архитектура программного продукта и структура базы данных должны быть разработаны с учетом требований ГОСТ 7.1–2003, а именно: с учетом наличия определенных областей библиографического описания и элементов этих областей. Причем, необходимо учесть, что ряд элементов в библиографическом описании могут повторяться (например, сведения об ответственности, место издания, примечание и др.).

2. Должна быть обеспечена возможность хранения в базе данных библиографических записей о книгах, статьях, электронных ресурсах, научных конференциях и т. д.

Обоснование. Указанный ГОСТ является основным стандартом, регламентирующим создание библиографических описаний любых объектов: книг, журналов, диссертаций, материалов конференций, статей, нот, карт, рукописей, патентных документов, электронных ресурсов и т. д.

Ссылки на спецификацию. Пп. 3.1, 3.9.

ПФТ2. Ввод библиографических описаний в базу данных в готовом виде

1. Библиографические описания должны вводиться в базу данных сразу в готовом виде, а не формироваться из элементарных фрагментов библиографического описания, как это делается в «больших» АБИС (например, «ИРБИС»). Таким образом, библиографическое описание формируется пользователем, а не программным продуктом.

2. Должна быть предусмотрена возможность ввода в базу данных детальных сведений, соответствующих областям и элементам областей библиографического описания (заглавие, сведения об ответственности и т. д.). Детальные сведения могут быть введены позднее, если пользователь нуждается в этом, но могут и не вводиться совсем.

Обоснование. Предлагаемая схема действий является инверсией по отношению к традиционной схеме. Такая схема позволит реализовать различные способы использования программного продукта. Простейший способ предусматривает только ввод библиографических описаний в целом, без ввода детальных данных по каждой области библиографического описания. Такой подход упрощает и ускоряет работу на начальном этапе, но в этом случае пользователь не сможет выполнять целый ряд поисковых операций в базе данных.

Ссылки на спецификацию. П. 3.9.

ПФТ3. Обработка наличия различных вариантов имен собственных

1. База данных должна быть спроектирована таким образом, чтобы наличие различных вариантов написания фамилий персоналий и наименований организаций, переименований городов и т. п. не препятствовало однозначной идентификации таких объектов пользователем.

2. Пользователь должен иметь возможность при выборе любого из вариантов, представленных в базе данных, получить выборку, включающую и все другие варианты.

Обоснование. Подобные разночтения фамилий персоналий могут появиться, например, при переводе иностранных книг на русский язык (*Вильямс* и *Уильямс*). Поэтому если пользователь производит поиск книг автора по фамилии *Вильямс*, то в выборку должны быть включены и книги этого же автора, но в переводе представленного как *Уильямс*, а в оригинале – как *Williams*. При этом в выборку не должны попасть книги однофамильцев, поскольку это уже различные персоналии.

Ссылки на спецификацию. Пп. 3.7, 3.9.

ПФТ4. Фиксирование результатов проработки источников

1. Должна быть предусмотрена возможность фиксирования даты и результатов проработки источников (книг, журналов, статей и т. д.).

2. Должна быть предусмотрена возможность выставления оценок проработанным источникам. Пользователь должен иметь возможность сам назначить показатели, по которым выставляются эти оценки.

3. Пользователь должен иметь возможность ввести в базу данных содержание (оглавление) проработанного источника.

Обоснование.

Ссылки на спецификацию. П. 3.9.

ПФТ5. Хранение выписок из проработанных источников

1. Должна быть предусмотрена возможность хранения выписок в базе данных.

2. Должна быть предусмотрена возможность связывания таких выписок, которые могут храниться также и в файлах, имеющих различные форматы (Microsoft Word, HTML, ASCII), с соответствующими библиографическими описаниями.

Обоснование. Необходимо избавить пользователя от утомительного поиска нужного файла методом визуального просмотра каталогов операционной системы (как зачастую бывает в действительности).

Ссылки на спецификацию. П. 3.9.

ПФТ6. Система ключевых слов

1. Должна быть предусмотрена система ключевых слов.

2. Должна быть предусмотрена возможность проведения выборки библиографических описаний для каждого ключевого слова.

Обоснование.

Ссылки на спецификацию. П. 3.9.

ПФТ7. Простые средства поиска информации

1. Должны быть предусмотрены простые средства поиска в базе данных и выборки информации из нее. На последующих этапах разработки эти средства должны быть значительно усилены.

Обоснование.

Ссылки на спецификацию. Пп. 3.2, 3.7.

ПФТ8. Учет экземпляров

1. Должна быть предусмотрена возможность описания экземпляров книг, журналов, дисков, рукописей и т. д., находящихся как в собственности пользователя, так и в библиотеках или у других лиц.

Обоснование.

Ссылки на спецификацию. П. 3.9.

5.2. Дополнительные функции для первой версии

ПФТ9. Формирование тематических библиографических списков

1. Должна быть предусмотрена возможность формирования библиографических списков, включающих описания группы объектов (книг, статей и т. д.), содержащихся в базе данных.

2. Списки должны формироваться без повторного ввода библиографических описаний в базу данных (без дублирования данных в ней).

Обоснование. Такие списки могут включать, например, литературу, предназначенную для выполнения одной задачи или характеризующую одну тему, или относящуюся к одному периоду времени (статьи за конкретный год по конкретной тематике) и т. п. При подготовке научной статьи или реферата ученый или студент сможет включить в такой список соответствующие источники и затем перенести сформированный список в статью или реферат в качестве перечня цитируемых источников.

Ссылки на спецификацию. П. 3.9.

ПФТ10. Развитые средства помощи пользователю для ускорения ввода данных

1. Должен быть создан механизм для ускоренного ввода типовых фрагментов текста в поля библиографического описания.

Обоснование. Такими типовыми фрагментами могут быть: фамилии авторов, сведения, относящиеся к заглавию, названия издательств, тексты примечаний и т. д.

Ссылки на спецификацию. П. 3.2, 3.7.

5.3. Будущие функции

ПФТ11. Развитые средства поиска информации

1. Должны быть предусмотрены различные средства поиска в базе данных и выборки информации из нее по различным критериям.

Обоснование.

Ссылки на спецификацию. П. 3.2, 3.7.

ПФТ12. Иерархическая система ключевых слов

1. Система ключевых слов должна быть организована по иерархическому принципу.

2. Должна быть предусмотрена возможность выполнения поиска информации в соответствии с требуемым уровнем иерархии ключевых слов.

Обоснование. Реализация этого требования позволит пользователю регулировать объем выбираемых библиографических описаний и точность соответствия запросу.

Например, для ключевого слова «Linux» словом-родителем будет ключевое слово «операционная система», а для него, в свою очередь, таким словом-родителем будет «система». Отметим, что в известных нам программных продуктах, используемых в библиотеках, иерархический подход к организации системы ключевых слов не реализован.

Ссылки на спецификацию. П. 3.7, 3.9.

ПФТ13. Формирование библиографического описания из элементарных данных

1. Должен быть реализован и традиционный способ алгоритмического формирования библиографического описания из элементарных данных, предварительно введенных пользователем в области и элементы библиографического описания (заглавие, сведения об ответственности, издатель, место издания, год издания и т. д.).

Обоснование. Реализация этого требования позволит опытному пользователю ускорить процесс ввода сведений в базу данных при полномасштабном использовании программного продукта.

Ссылки на спецификацию. П. 3.7.

Функция	Приоритет	Уровень трудозатрат	Риск	Целевая версия
ПФТ1	критический	средний	низкий	1.0
ПФТ10	важный	высокий	средний	1.0
...	...			
ПФТ13	полезный	средний	низкий	3.0

6. Основные способы использования программного продукта и сценарии работы с ним

6.1. Простой способ использования программного продукта

Пользователь вводит в базу данных только библиографические описания (полные и краткие либо только краткие). Он не использует систему справочников. Он не вводит в базу данных детальные сведения, содержащиеся в областях библиографического описания (заглавие, сведения об ответственности, издатель, место издания, год издания и т. д.). В таком случае пользователь лишен возможности выполнять большую часть процедур поиска сведений в базе данных.

6.2. Полномасштабный способ использования программного продукта

Пользователь использует систему справочников, дополняет их новыми данными по мере необходимости. Он вводит в базу данных полные и краткие библиографические описания (либо только краткие, если описание выполняется не на основе оригинального

объекта, а на основе вторичной информации, например, библиографического списка в книге, диссертации, статье). Он вводит в базу данных детальные сведения, содержащиеся в областях библиографического описания (заглавие, сведения об ответственности, издатель, место издания, год издания и т. д.). Также пользователь ведет учет экземпляров (книг, журналов, дисков, рукописей и т. д.), как находящихся в собственности пользователя, так и находящихся в библиотеках или у других лиц.

7. Нефункциональные требования

7.1. Требования к программному продукту

7.1.1. Требования к инсталляции

Необходимо разработать процедуру инсталляции программного продукта.

7.1.2. Требования к эксплуатации

7.1.2.1. Требования к удобству эксплуатации (практичность)

7.1.2.1.1. Необходимое время подготовки пользователя для достижения минимальной производительности

Приблизительно 1–2 дня. Пользователю необходимо ознакомиться с ГОСТ 7.1–2003. Он может ограничиться только просмотром примеров библиографических описаний, приведенных в Приложении к этому ГОСТу.

7.1.2.1.2. Время выполнения типичных задач или транзакций, осуществляемых пользователем

Ввод полного и краткого библиографических описаний для одной книги (статьи, диска и т. д.) должен занимать не более 10 минут. Ввод дополнительных сведений, в соответствии с областями библиографического описания, должен занимать не более 1 часа. По мере повышения квалификации пользователя эти временные интервалы должны уменьшиться до 5 минут и 20 минут соответственно. Это должно быть достигнуто за счет того, что по мере приобретения навыков работы с программным продуктом пользователь должен постепенно заполнить различные справочники, содержащиеся в базе данных, а также настроить конфигурационный файл, содержащий типовые фрагменты библиографических описаний (имена и отчества авторов, наименования часто встречающихся издательств и т. п.).

7.1.2.1.3. Сравнение практичности новой системы с уже существующими современными системами

Практичность должна быть не хуже, чем у всех систем, перечисленных в документе-концепции.

7.1.2.1.4. Следование соглашениям и стандартам, разработанным для человеко-машинного интерфейса

Пользовательский интерфейс программного продукта должен отвечать, как минимум, требованиям здравого смысла. Хотя это требование неконкретное, но, поскольку автором документа-концепции и разработчиком будет одно и то же лицо, а также в силу ограниченности времени на разработку, особых требований относительно соответствия стандартам, разработанным для человеко-машинного интерфейса, на данном этапе не предъявляется. В дальнейшем возможны изменения технологии создания пользовательского интерфейса, например, использование каскадных таблиц стилей (CSS).

7.1.2.1.5. Прочие требования к удобству эксплуатации

ПНТ1. Экспорт и импорт данных

1. Должна быть реализована возможность экспорта данных из системы в виде команд языка SQL и импорта данных, представленных в виде таких команд.

Обоснование. Это позволит организовать обмен данными между пользователями предлагаемой системы. Поскольку оформление библиографических записей представляет собой трудную задачу для большинства студентов и аспирантов, то в процессе заполнения индивидуальной базы данных каждого пользователя важную роль может сыграть взаимный обмен данными. В частности, преподаватель, рекомендуя студентам перечень источников для изучения материала конкретной учебной дисциплины, мог бы выдавать им этот перечень сразу в той форме, которая позволяет выполнить его импорт в базу данных каждого студента.

Ссылки на спецификацию. П. 3.10.

ПНТ2. Конфигурирование программного продукта

1. Должна быть предусмотрена возможность конфигурирования программного продукта.

2. Конфигурационные параметры должны описывать, в основном, значения полей документов, принимаемые по умолчанию и позволяющие пользователю ускорить процесс ввода данных.

3. В первой версии программного продукта реализация конфигурирования может быть выполнена на основе использования текстового конфигурационного файла, изменения в который должны вноситься пользователем в текстовом редакторе.

4. В последующих версиях должна быть разработана интерактивная система конфигурирования программного продукта.

Обоснование. Реализация этого требования позволит пользователю ускорить процесс ввода данных, сделать его более удобным и уменьшить число ошибок ввода.

Ссылки на спецификацию. П. 3.8.

7.1.2.2. Требования к режиму эксплуатации

7.1.2.2.1. Доступность (сколько дней в неделю и часов в сутки должна быть доступна система)

Эпизодически, по мере возникновения потребности.

7.1.2.2.1. Прочие требования к режиму эксплуатации

Нет требований.

7.1.2.3. Прочие требования к эксплуатации

Нет требований.

7.1.3. Требования к эффективности

7.1.3.1. Требования к производительности

7.1.3.1.1. Время ответа для транзакции (среднее, максимальное)

Время получения результатов при выполнении выборки библиографических описаний не должно превышать 2–3 секунды.

7.1.3.1.2. Пропускная способность (число транзакций в секунду)

Нет требований.

7.1.3.1.3. Емкость (число пользователей или транзакций, которые может обслужить система)

Нет требований.

7.1.3.1.4. Режимы снижения производительности (допустимые режимы работы при ухудшении параметров системы)

Нет требований.

7.1.3.1.5. Время обновления экрана

Нет требований.

7.1.3.1.6. Время реакции на действия пользователя

Нет требований, т. к. все операции по вводу и корректировке данных не связаны с обработкой больших объемов данных.

7.1.3.2. Требования к ресурсам

7.1.3.2.1. Клиентский компьютер

Оперативная память: объем, достаточный для нормальной работы Web-браузера. Объем дискового пространства: нет требований. Процессор: любой современный процессор.

7.1.3.2.2. Сервер

Оперативная память: объем, достаточный для нормальной работы Web-сервера и сервера баз данных. Объем дискового пространства: приблизительно 100–150 Мб с учетом СУБД PostgreSQL (дополнительный требуемый объем зависит от объема электронных документов). Процессор: определяется требованиями, которые предъявляет СУБД PostgreSQL (см. документацию на текущую версию этой СУБД).

7.1.3.3. Прочие требования к эффективности

Нет требований.

7.1.4. Требования к надежности

7.1.4.1. Вероятность отказа

Нет требований, т. к. от программного продукта не требуется непрерывное функционирование.

7.1.4.2. Частота отказов

Нет требований, т. к. от программного продукта не требуется непрерывное функционирование.

7.1.4.3. Среднее время безотказной работы (среднее время между двумя последовательными сбоями)

Не менее отрезка времени, требующегося для ввода данных об одном объекте библиографического описания (книге, статье и т. д.).

7.1.4.4. Вероятность готовности системы к использованию

Нет требований, т. к. от программного продукта не требуется непрерывное функционирование.

7.1.4.5. Среднее время восстановления после сбоя (отказа)

Нет требований, т. к. программный продукт не является критическим по данному показателю.

7.1.4.6. Вероятность порчи данных при сбое (отказе)

Нет требований, т. к. предполагается, что СУБД обеспечивает откат незавершенных транзакций, гарантируя тем самым целостность базы данных.

7.1.4.7. Прочие требования к надежности

Нет требований.

7.1.5. Требования к переносимости

7.1.5.1. Поддерживаемые операционные системы

7.1.5.1.1. Клиентский компьютер

Операционные системы UNIX (FreeBSD, Linux) и Windows (XP/2003/Vista/7). На клиентском компьютере работает Web-браузер.

7.1.5.1.2. Сервер

Операционная система UNIX (FreeBSD, Linux). В будущем предполагается перенос продукта в среду операционной системы Windows (XP/2003/Vista/7).

7.1.5.2. Объем машинно-зависимых операторов и подсистем

Таких операторов и подсистем нет.

7.1.5.3. Прочие требования к переносимости

Нет требований.

7.1.6. Требования к защищенности

ПНТ3. Аутентификация пользователей

1. При входе пользователя в систему должна выполняться аутентификация.

Обоснование. Это необходимо для организации разграничения полномочий доступа к базе данных.

Ссылки на спецификацию. Пп. 3.1, 3.9.

ПНТ4. Интерактивная система управления разграничением полномочий доступа к базе данных

1. Должна быть разработана интерактивная система для управления разграничением полномочий доступа различных пользователей к базе данных.

Обоснование.

Ссылки на спецификацию. Пп. 3.5, 3.9.

ПНТ5. Ведение журнала операций с базой данных

1. Должна быть предусмотрена возможность ведения журнала всех операций с базой данных, изменяющих ее состояние: ввод записей, их обновление и удаление.

2. Должна быть предусмотрена возможность просмотра истории всех операций с базой данных с применением специального пользовательского интерфейса.

Обоснование. В случае ошибочного изменения или удаления данных пользователь будет иметь возможность «реанимировать» их.

Ссылки на спецификацию. П. 3.9.

7.1.7. Требования к безопасности

При инсталляции программного продукта необходимо сохранять резервные копии всех конфигурационных файлов, принадлежащих другим программным продуктам, в случае внесения изменений в эти файлы.

7.1.8. Требования к удобству сопровождения

Должны быть подробные комментарии в исходных текстах программ.

7.1.9. Требования к точности числовых вычислений

Нет требований.

7.1.10. Прочие требования к программному продукту

ПНТ6. Организация коллективной работы

1. Должна быть предусмотрена возможность коллективного использования программного продукта через локальную сеть или Internet, а также на одном компьютере.

2. Должен быть организован отдельный доступ к данным, принадлежащим различным пользователям.

Обоснование.

Ссылки на спецификацию. Пп. 3.1, 3.9.

7.2. Организационные требования

7.2.1. Выходные требования

7.2.1.1. Сроки разработки и изготовления

Первый работающий вариант программного продукта должен быть готов к 15 сентября 2006 г.

7.2.1.2. Сопутствующая документация

Нет сопутствующей документации.

7.2.1.3. Прочие выходные требования

Нет требований.

7.2.2. Требования к реализации

7.2.2.1. Модель организации разработки

Модель пошаговой разработки.

7.2.2.2. Методы проектирования и документирования разработки

Нет требований.

7.2.2.3. Языки программирования и инструментальные средства

1. Языки – Perl (версия 5.8.x) и JavaScript; СУБД – PostgreSQL (версия 8.x); Web-сервер – Apache (версия 2.2.x). Инструментальные средства: отладчик языка Perl; отладчик языка JavaScript, встроенный в Web-браузер. Библиотеки: интерфейс к СУБД PostgreSQL – pgperl (до версии 7.3 входил в состав дистрибутива СУБД PostgreSQL).

2. Рассмотреть также возможность использования системы управления версиями программного продукта (Subversion, Git и др.).

3. Рассмотреть возможность использования общего интерфейса DBI/DBD к СУБД PostgreSQL.

7.2.2.4. Локализация и интернационализация

Необходимо реализовать поддержку локализации (l10n) и интернационализации (i18n) на более поздних этапах разработки. Поддерживаемые языки: русский и английский.

7.2.3. Требования к используемым стандартам

1. Использовать стандарты кодирования и оформления исходных текстов программ, аналогичные стандартам проекта GNU.
2. Комментарии в исходных текстах программ должны быть написаны на русском языке.

7.2.4. Требования к лицензированию

Программный продукт должен распространяться по лицензии, аналогичной лицензии GNU GPL.

7.2.5. Требования к дистрибуции (распространению) и вопросы цены

1. Распространение продукта будет производиться, в основном, через Web-сайт <http://www.morgunov.org>.
2. Программный продукт будет распространяться бесплатно в исходных текстах.

7.2.6. Вопросы авторских прав

Авторские права принадлежат Моргунову Е. П.

7.3. Внешние требования

7.3.1. Требования к взаимодействию с другими системами

1. Необходимо организовать взаимодействие с СУБД PostgreSQL, которая будет использоваться для хранения данных, и с Web-сервером Apache.
2. Взаимодействие с другими *прикладными* программными продуктами не предполагается.

7.3.2. Юридические требования

Для разработки программного продукта должны использоваться только инструменты с лицензией, разрешающей их бесплатное использование.

7.3.3. Этические требования

Нет требований.

7.4. Прочие требования

Нет требований.

8. Требования к документации

8.1. Руководство пользователя

1. Должно содержать:
 - основы методики составления библиографического описания (на основе ГОСТ 7.1–2003);
 - примеры библиографических описаний;
 - описание процедур работы с продуктом;
 - глоссарий;
 - индекс для поиска по тексту.
2. Форма руководства – электронная, формат – pdf, html. Должна быть предусмотрена возможность вывода руководства на печать (полностью или отдельных разделов).

8.2. Руководство программиста

- Должно содержать следующие сведения (но не обязательно ограничиваться ими):
- описание порядка установки программного продукта и его первичной настройки;
 - описание модулей программы и их взаимосвязей;
 - описание типовой структуры модуля и порядка его функционирования.

8.3. Интерактивная подсказка

1. Должна быть контекстно-зависимой и иметь развитую систему ссылок и функцию поиска по тексту.
2. Должна быть в состоянии предложить пользователю подсказки по заполнению полей экранных форм.

8.4. Руководства по установке и конфигурированию и файл ReadMe

Должны быть представлены в электронной форме (формат файла – ASCII). Файл ReadMe должен содержать перечень изменений, проведенных в новой версии, а также список замеченных ошибок.

8.5. Маркировка и упаковка

В процессе установки необходимо выводить информацию о версии программного продукта и о его авторе. При распространении продукта на CD-дисках предусмотреть наличие маркировки с информацией о наименовании и версии продукта, а также сведений об авторе.

9. Глоссарий и список сокращений

9.1. Глоссарий

9.2. Список сокращений

АБИС – автоматизированная библиотечная информационная система

ББК – библиотечно-библиографическая классификация

БД – база данных

ГПНТБ – Государственная публичная научно-техническая библиотека России

ГРНТИ – Государственный рубрикатор научно-технической информации

ОС – операционная система

ПНТ – пользовательское нефункциональное требование

ПФТ – пользовательское функциональное требование

СИБИД – Система стандартов по информации, библиотечному и издательскому делу

СУБД – система управления базами данных

УДК – универсальная десятичная классификация (система классификации информации, широко используется во всем мире для систематизации произведений науки, литературы и искусства, периодической печати, различных видов документов и организации картотек)

ЮНЕСКО (англ. UNESCO – United Nations Educational, Scientific and Cultural Organization) – Организация Объединённых Наций по вопросам образования, науки и культуры

CSS – Cascading Style Sheets (каскадные таблицы стилей)

CSV – Comma separated values (текстовый формат, предназначенный для представления табличных данных)

GNU – рекурсивный акроним от англ. GNU's Not UNIX (проект, основанный Ричардом Столлманом – <http://www.gnu.org>)

GNU GPL – GNU General Public License (лицензия на свободное программное обеспечение, созданная в рамках проекта GNU)

I18n – internationalization (интернационализация)

ISBN – International Standard Book Number (международный стандартный номер книги – уникальный номер книжного издания, необходимый для распространения книги в торговых сетях и автоматизации работы с изданием)

L10n – localization (локализация)

Задание 7

Задание выполняется в группе из 3-4 человек

Разработка спецификаций

Рекомендации

Пункт 4.1 нужно писать так, чтобы если бы эту задачу поручили вам, то вам будет понятно, что нужно сделать.

Если пунктов в 4.1 много, можно сделать подробную спецификацию только пяти из них. В остальных указать ...

В пункте 4.2 делайте реалистичные предположения.

Разделите пункты спецификации между участниками команды для ускорения работы.

Неиспользуемые пункты верхнего уровня (с номером) оставьте, но укажите "нет" или "не требуется".

Шаблон спецификаций

Содержание

1. Введение

1.1 Цель

В данном разделе нужно указать цель данной SRS, которая должна полностью описывать внешнее поведение конкретного приложения или подсистемы, а также нефункциональные требования, ограничения проектирования и другие элементы, необходимые для обеспечения всестороннего описания требований к программному обеспечению,

1.2. Масштаб

Данный раздел содержит краткое описание программного приложения (функций и/или подсистем, на которые разбита система), для которого создается спецификация; кроме того, описывается, с какой моделью (моделями) вариантов использования оно связано, а также все остальное на что оказывает влияние данный документ.

1.3. Ссылки.

Список ссылок или прилагаемых документов, связанных с данным проектом.

1.4. Предположения и зависимости

В данном разделе описывается техническая достижимость, доступность подсистем или компонентов и другие предположения, от которых может зависеть жизнеспособность описываемого данной SRS программного обеспечения.

2. Краткая характеристика модели вариантов использования

Данный раздел содержит краткую характеристику модели вариантов использования. Она предназначена для тех, кто интересуется поведением системы, — заказчиков, пользователей, архитекторов, авторов вариантов использования, разработчиков, разработчиков вариантов использования, тестологов, менеджеров, ревизоров и авторов документации. Для каждого варианта использования необходимо указать следующее.

- Название варианта использования.
- Краткое описание, объясняющее функцию варианта использования и его роль в системе.
- Перечень акторов данного варианта использования.
- Диаграмма модели вариантов использования. (Здесь следует поместить диаграмму модели вариантов использования в целом.)

3. Характеристика акторов

Здесь описываются все упомянутые в характеристике модели вариантов использования акторы. Для каждого актора следует указать следующее. * Имя * Краткое описание

4. Требования

4.1. Функциональные требования

В данном разделе описываются функциональные требования к системе, выраженные на естественном языке. Для многих приложений это достаточно объемная информация, и следует продумать, как организовать данный раздел. Как правило, его организуют по функциям, но можно применять и другие методы, например по пользователям или подсистемам. При использовании для сбора функций вспомогательных средств разработки приложений (инструментальных средств разработки требований, средств моделирования и т.д.) данный раздел документа будет содержать ссылки на эти данные и указывать местоположение и название применяемого для сбора данных инструментального средства.

4.2. Нефункциональные требования

Большая часть нефункциональных требований обычно записывается на естественном языке в данном разделе спецификации. Но нефункциональные требования могут также входить в спецификации вариантов использования.

4.2.1. Практичность

В данный раздел следует включить все требования, влияющие на практичность программного обеспечения. Как правило, указывается следующее.

*Время, необходимое для обучения рядовых пользователей и пользователей с большими полномочиями, чтобы они научились эффективно выполнять определенные действия. *
Время выполнения типичных задач; или же практичность новой системы, сравнивается с практичностью известных систем, которые пользователь знает и любит. * Требования соответствия общепринятым стандартам практичности, таким как CUA IBM или опубликованные компанией Microsoft стандарты GUI для системы Windows 98.

4.2.2. Надежность

В данном разделе указываются требования к надежности системы.

- Доступность. Указывается, какой процент времени система доступна (xx.xx%), определяются часы использования и доступа для обслуживания, операции при ухудшении параметров системы и т.д.
- Среднее время между отказами (mean time between failures, MTBF). Обычно выражается в часах, но может указываться в днях, месяцах и годах.
- Среднее время восстановления (mean time to repair, MTTR) Сколько времени система может находиться в нерабочем состоянии после сбоя.
- Точность. С помощью некоего известного стандарта указывается требуемая точность (разрешающая способность) выводимой системой информации.
- Максимально допустимый коэффициент ошибок и дефектов. Как правило, выражается как число ошибок, приходящееся на KLOS (тысячу строк кода), или число ошибок, приходящихся на отдельную функцию.
- Доля ошибок или дефектов различных типов. Обычно ошибки разбиваются на следующие категории: незначительные, серьезные и критические. Требования должны определять, что понимается под "критической" ошибкой (такой, как полная потеря данных или невозможность использовать определенную часть функциональных возможностей системы).

4.2.3. Производительность

Здесь описываются характеристики производительности системы. Следует указать время ответа для различных ситуаций. Если требуется, указываются названия соответствующих вариантов использования.

- Время ответа для транзакции (среднее, максимальное)
- Пропускная способность (транзакций в секунду)
- Емкость (число пользователей или транзакций, которые может обслужить система)
- Режимы снижения производительности (допустимые режимы работы при ухудшении параметров системы)
- Использование ресурсов (память, диск, каналы связи)

4.2.4. Возможность сопровождения

Данный раздел содержит требования, способствующие улучшению возможности сопровождения и обслуживания создаваемой системы, в том числе стандарты кодирования, определенные соглашения, библиотеки классов, доступ для обслуживания и вспомогательные обслуживающие программы.

5. Требования к интерактивной документации пользователя и системе подсказок

Здесь описываются требования (если таковые имеются) к интерактивной документации пользователя, системе подсказок и т.д.

6. Ограничения проектирования

В данном разделе следует описать все ограничения проектирования создаваемой системы. Ограничения проектирования представляют решения по проектированию, которые являются обязательными и должны быть выполнены. Например, может задаваться язык программирования, требования к программным процессам, а также может предписываться использование определенных средств разработки, архитектурных и проектных ограничений, закупаемых компонентов и библиотек классов.

7. Закупаемые компоненты

В этом разделе описываются все используемые в системе закупаемые компоненты и соответствующие ограничения лицензирования или использования, а также все связанные с ними стандарты совместимости/взаимодействия или интерфейсов.

8. Интерфейсы

В данном разделе определяются интерфейсы, которые должны поддерживаться приложением. Раздел должен содержать достаточно подробное описание протоколов, портов, логических адресов и т.п., чтобы можно было разработать программное обеспечение и проверить его соответствие налагаемым на интерфейсы требованиям.

8.1. Интерфейсы пользователя

Описываются интерфейсы пользователя, которые должны быть реализованы программным обеспечением.

8.2. Аппаратные интерфейсы

Определяются все аппаратные интерфейсы, поддержку которых должно осуществлять программное обеспечение, в том числе логическая структура, физические адреса и ожидаемое поведение.

8.3. Интерфейсы программного обеспечения

Описываются программные интерфейсы с другими компонентами системы программного обеспечения. Это могут быть закупаемые компоненты, повторно используемые компоненты другого приложения или компоненты, разработанные для подсистем, не описываемых данной SRS, но с которыми данное программное приложение должно взаимодействовать:

8.4. Коммуникационные интерфейсы

Описываются все коммуникационные интерфейсы с другими системами или устройствами, такими как локальные сети или удаленные последовательные порты.

9. Требования лицензирования

Определяются все требования лицензирования или другие ограничивающие использование требования, которые оказывают влияние на программное обеспечение.

10. Замечания, касающиеся законности, авторских прав и т.д.

Описываются все необходимые гарантии, все отказы от ответственности, отметки об авторском праве, торговой марке или вопросы соответствия логотипу для программного обеспечения.

11. Применяемые стандарты

Посредством ссылок указываются все стандарты (а также конкретные их разделы), которые применяются к описываемой системе. Например, это могут быть стандарты качества, некие законы или инструкции, а также отраслевые стандарты практичности, взаимодействия, интернационализации, соответствия операционной системы и т.д.

Индекс

Наличие индекса помогает читателю определять местонахождение в документе ключевых понятий и тем.

Глоссарий

Здесь описываются все термины данного приложения, а также все определения и принятые в проекте или компании сокращения, которые необходимы для понимания данного документа и приложения.

Приложения

Критерии оценки

- Структура спецификации соответствует шаблону - 2 балла
- Выполняется трассируемость функций из документа концепции - 1 балл
- Имеется подробная спецификация не менее 5 функциональных требований - 2 балла
- Имеется подробная спецификация не менее 3 нефункциональных требований - 2 балла
- Указана спецификация для интерфейсов пользователя (не менее 3 диалогов) - 2 балла
- Указана спецификация для хотя бы 1 интерфейса ПО - 1 балл

Пример

Тема: АРМ "Кафедра" Составление и заполнение индивидуального-учебного плана преподавателя, составление плана кафедры, выполнение плана кафедры. Печать отчетов. Связь со сторонними ИС.

SRS

1. Введение

1.1. Цель:

Разработать ПП «АРМ Кафедра», главная цель которого - автоматизация документооборота на кафедре, а именно автоматизация планирования работы кафедры и автоматизация отчетности работы преподавателя по ИПП.

1.2. Область применения

Использование данного продукта на любой кафедре как в ИрГУПС, так и в других высших и средних учебных заведениях.

1.3. Термины и сокращения

- SRS (Software Requirements Specifications) – спецификация требований к ПО;
- ИрГУПС – Иркутский Государственный Университет путей сообщения;
- ИС – Информационные системы;
- ПО – Программное обеспечение;
- ЛВС - Локальная вычислительная сеть;
- ВУЗ - Высшее учебное заведение;
- ЭВМ – Электронно-вычислительная машина;
- ЕБД – Единая база данных (*здесь: университета*);
- СУБД – Система управления базами данных;
- MS IE – Microsoft Internet Explorer;
- ИЕЕЕ - Института инженеров по электротехнике и радиоэлектронике;
- ИПП- Индивидуальный план преподавателя;
- ПК – План кафедры;
- ПП- программный продукт;
- Документы – Индивидуальный план преподавателя и план кафедры;
- Отчеты- Индивидуальный план преподавателя и план кафедры, представленные в том виде, в котором они должны быть на печати.

1.4. Ссылки

- **Положение** о планировании учебной, учебно-методической, научно-исследовательской, организационно-методической и воспитательной работы в ИрГУПС;

- **Орлов С.А.**

Технологии разработки программного обеспечения: разработка сложных программных систем. – 2-е изд. – СПб.: Питер, 2002. – 473с. – (Учебное пособие, в качестве дополнительного источника информации);

- Справочная информация по PHP:

www.php.net

- Справочная информация по MySQL:

www.mysql.com

- Справочная информация по Apache:

www.apache.com

- Стандарты IEEE:

www.ieee.org

- **Наварро Э.**

XHTML: учебный курс. – СПб.: Питер, 2001.-336с.:ил.

- **Гутманс Э., Бакен С., Ретанс Д.**

PHP5. Профессиональное программирование.- Пер. с англ.- СПб.: Символ-Плюс, 2006.-704с.:ил.

- **Рева О.Н.**

HTML. Просто как дважды два.-М.: Изд-во Эксмо, 2006.-256с.:ил.

1.5. Обзор продукта

Данный продукт является помощником в создании индивидуального учебного плана преподавателя и плана кафедры. С его помощью можно будет заполнить, отредактировать, просмотреть и распечатать план. Данная система будет установлена на кафедре и даст возможность работающему на кафедре персоналу, в зависимости от прав пользователей, осуществлять некоторые действия с планом.

Продукт должен обладать простым и удобным интерфейсом, в котором должны быть учтены все требования пользователя.

Основные функции:

- Автоматическое сопоставление часов в планах преподавателей и в плане кафедры;
- Возможность создания и редактирования планов;
- Выполнение планов;
- Вывод отчета в текстовый процессор (MSWord 2003, OpenOffice);
- Печать отчетов;

2. Общее описание

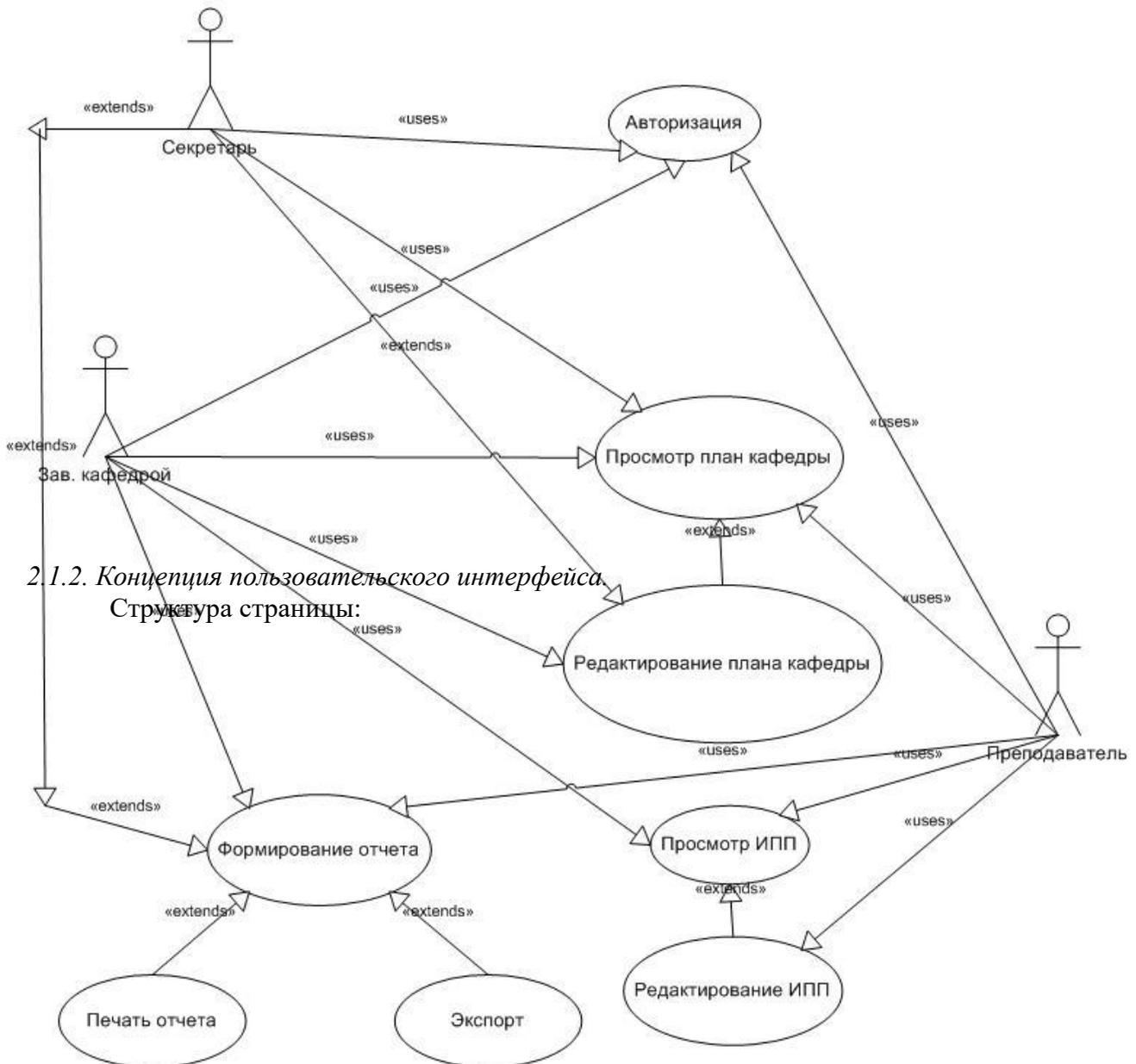
2.1. Перспективы продукта

Данный продукт может быть использован на любой кафедре университета. Так как предполагает универсальный интерфейс, и связь с единой базой данных университета.

2.1.1. Концепция операций

Диаграмма прецедентов (Use Case)

USE CASE 1



2.1.2. Концепция пользовательского интерфейса

Структура страницы:



Верхнее меню – содержит ссылки на различные функции ПП;

Область заголовка – отображают логотип системы, название системы и аутентификации пользователя;

Боковое меню – отображает доступные действия;

Рабочая область – отображает информацию в соответствии с последними действиями пользователя.

Нижний колонтитул – содержит информацию о разработчиках.

2.1.3. Аппаратные интерфейсы.

Стандартное оборудование компьютера, включающее монитор, клавиатуру, мышь, модем.

2.1.4. Программные интерфейсы.

Наличие web-браузера.

2.1.5. Коммуникационный интерфейс.

Для коммуникации и передачи данных необходима компьютерная сеть с доступом к WEB-серверу, на котором расположен ПП, по протоколу HTTP.

2.1.7. Операции

Данный продукт состоит из базы данных и пользовательских интерфейсов. Составление планов будет производиться следующим образом:

- Инициализация;
- Вход в систему;
- Просмотр плана (кафедры или индивидуального плана преподавателя);

- Редактирование имеющейся таблицы (некоторая часть таблицы заполняется в ручную);
- Формирование отчета;
- Экспорт отчета в текстовый процессор;
- Печать;

2.2 Функции продукта:

1. **Просмотр** постранично плана кафедры или плана преподавателя при нажатии на соответствующие пункты меню; автоматически открывается план кафедры, к которой принадлежит сотрудник, для преподавателя открывается его план.
2. **Редактирование** документа, в зависимости от прав пользователя, будет осуществляться в так называемом – режиме редактирования. Пользователь может редактировать документ отдельные ячейки таблицы;
3. **Печать** производится из приложения по нажатии кнопки «печать». После нажатия кнопки происходит формирование отчета и возможность просмотреть отчет перед печатью. Распечатать можно как текущую страницу, так и весь документ целиком;
4. **Формирование отчета** производится после нажатия кнопок «Печать» или «Экспорт», при этом все формы документа, включая титульный лист, представлены для просмотра;
5. **Экспорт** отчета в текстовый процессор (перевод отчета по нажатию соответствующей кнопки в указанный формат, для работы с отчетом в MS Word, Open Office и др.) производится из приложения по нажатии кнопки «Экспорт». После нажатия кнопки происходит формирование отчета и возможность просмотреть отчет перед экспортом;

2.3 Пользовательские характеристики продукта

Типичными пользователя продукта будут люди, работающие на кафедре, обладающие знаниями о планировании учебной, учебно-методической, научно-исследовательской, организационно-методической и воспитательной работы в ВУЗе.

2.4 Ограничения

В плане кафедры в таблице №2 не отображается стаж, ученая степень, ученое звание профессорско-преподавательского штата кафедры, так как данной информации в ЕБД нет.

2.6 Распределения требований

Приоритеты требований к системе:

1. Сетевая система;
1. Обеспечение связи со сторонними ИС;
1. Составление и заполнение планов;
2. Многопользовательская система;
3. Дружественный интерфейс;
4. Возможность вывода отчета в текстовый редактор;
5. Модифицируемость системы;

3. Детальные требования

3.1 Требования к внешнему интерфейсу

3.1.1 Пользовательские интерфейсы

Макеты интерфейсов пользователей:

ЕИС-Кафедра

Здравствуйте, Иванов Иван Иванович

АРМ КАФЕДРА

ПОМОЩЬ ПЕЧАТЬ ЭКСПОРТ РЕДАКТИРОВАНИЕ ВЫХОД

ПЛАН ПРЕПОДАВАТЕЛЯ

- Учебная нагрузка
- Учет выполнения
- Учебно-методическая работа
- Научно-исслед. работа
- Орг.-методическая работа
- Воспитательная нагрузка со студентами
- Сводная статистика

ПЛАН КАФЕДРЫ

- Годовая нагрузка и штат кафедры
- Профессорско-преподавательский штат кафедры
- План заседаний
- Учебно-методическая работа кафедры
- Мероприятия кафедры по сотрудничеству с производством
- Развитие и совершенствования уч. лаб. базы
- Работа над повышением квалификации
- Контроль качества учебного процесса
- Перечень НИ тематики
- Диссертации
- Работа со студентами
- Дополнительные мероприятия кафедры

ПЛАН КАФЕДРЫ

План заседаний

№ п/п	Вопросы, назначенные к обсуждению на заседании кафедры	Фамилия докладчика	Дата	Отметка об исполнении
1.1	Подведение итогов работы кафедры ИС в 2007-2008 учебном году	Ермаков А.А.	Сентябрь 2008г.	
1.2	Задачи кафедры ИС на новый учебный год	Ермаков А.А.		
1.3	Отчет аспирантов кафедры ИС (аттестация)	Аспиранты		
2.1	Обсуждение и утверждение тематики дипломного проектирования на 2008-2009 учебный год	Бутырин О.В. Ермаков А.А.	Октябрь 2008г.	
2.2	О ходе выполнения учебно-методической работы	Бутырин О.В.		
2.3	О ходе выполнения редакционно-издательской работы	Бутырин О.В.		
3.1	Состояние УМК дисциплин кафедры ИС	Абасова Н.И.	Ноябрь 2008г.	

1. Интерфейс пользователя – преподаватель.

ЕИС-Кафедра

Здравствуйте, Иванов Иван Иванович

АРМ КАФЕДРА

ПОМОЩЬ ПЕЧАТЬ ЭКСПОРТ РЕДАКТИРОВАНИЕ ВЫХОД

ПЛАН КАФЕДРЫ

- Годовая нагрузка и штат кафедры
- Профессорско-преподавательский штат кафедры
- План заседаний
- Учебно-методическая работа кафедры
- Мероприятия кафедры по сотрудничеству с производством
- Развитие и совершенствования уч. лаб. базы
- Работа над повышением квалификации
- Контроль качества учебного процесса
- Перечень НИ тематики
- Диссертации
- Работа со студентами
- Дополнительные мероприятия кафедры

ПЛАН КАФЕДРЫ

План заседаний

№ п/п	Вопросы, назначенные к обсуждению на заседании кафедры	Фамилия докладчика	Дата	Отметка об исполнении
1.1	Подведение итогов работы кафедры ИС в 2007-2008 учебном году	Ермаков А.А.	Сентябрь 2008г.	
1.2	Задачи кафедры ИС на новый учебный год	Ермаков А.А.		
1.3	Отчет аспирантов кафедры ИС (аттестация)	Аспиранты		
2.1	Обсуждение и утверждение тематики дипломного проектирования на 2008-2009 учебный год	Бутырин О.В. Ермаков А.А.	Октябрь 2008г.	
2.2	О ходе выполнения учебно-методической работы	Бутырин О.В.		
2.3	О ходе выполнения редакционно-издательской работы	Бутырин О.В.		
3.1	Состояние УМК дисциплин кафедры ИС	Абасова Н.И.	Ноябрь 2008г.	

2. Интерфейс пользователя – секретарь.

Здравствуйте, Иванов Иван Иванович

АРМ КАФЕДРА

Иванов Иван Иванович (ст. преп.) ПЛАН КАФЕДРЫ

ПЛАН ЗАСЕДАНИЙ

№ п/п	Вопросы, назначенные к обсуждению на заседании кафедры	Фамилия докладчика	Дата	Отметка об исполнении
1.1	Подведение итогов работы кафедры ИС в 2007-2008 учебном году	Ермаков А.А.	Сентябрь 2008г.	
1.2	Задачи кафедры ИС на новый учебный год	Ермаков А.А.		
1.3	Отчет аспирантов кафедры ИС (аттестация)	Аспиранты		
2.1	Обсуждение и утверждение тематики дипломного проектирования на 2008-2009 учебный год	Бутырин О.В. Ермаков А.А.	Октябрь 2008г.	
2.2	О ходе выполнения учебно-методической работы	Бутырин О.В.		
2.3	О ходе выполнения редакционно-издательской работы	Бутырин О.В.		
3.1	Состояние УМК дисциплин кафедры ИС	Абасова Н.И.	Ноябрь 2008г.	

3. Интерфейс пользователя - заведующий кафедры.

3.1 Аппаратные интерфейсы

- Используется стандартное мультимедийное оборудование компьютера, включающее монитор, клавиатуру, мышь, модем.
- Использование ЭВМ с выходом в ЛВС.

3.1.3 Программные интерфейсы:

- Использование для обмена данными протокола SOAP;
- Используемая при разработке СУБД – MySQL;
- Взаимодействие с ЕБД.

3.1.4 Коммуникационные интерфейсы:

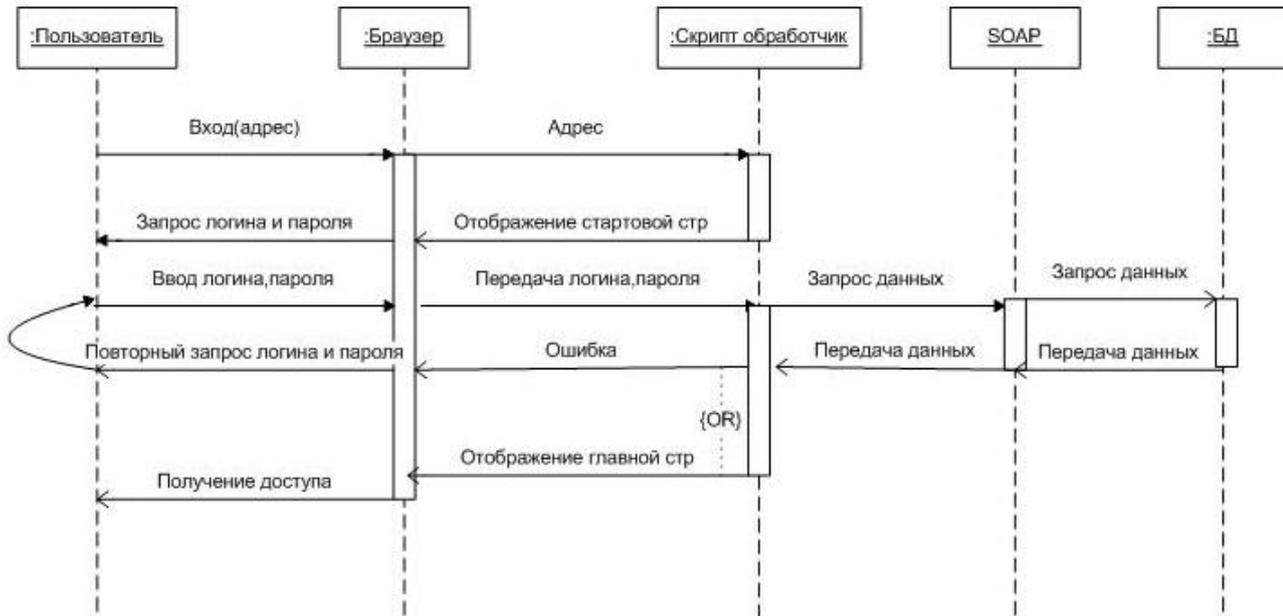
- Web-браузер (MS IE, FireFox, Opera);
- Возможно использование протоколов HTTP, FTP.

3.2. Детальные требования

3.2.1. Диаграммы последовательностей для операций:

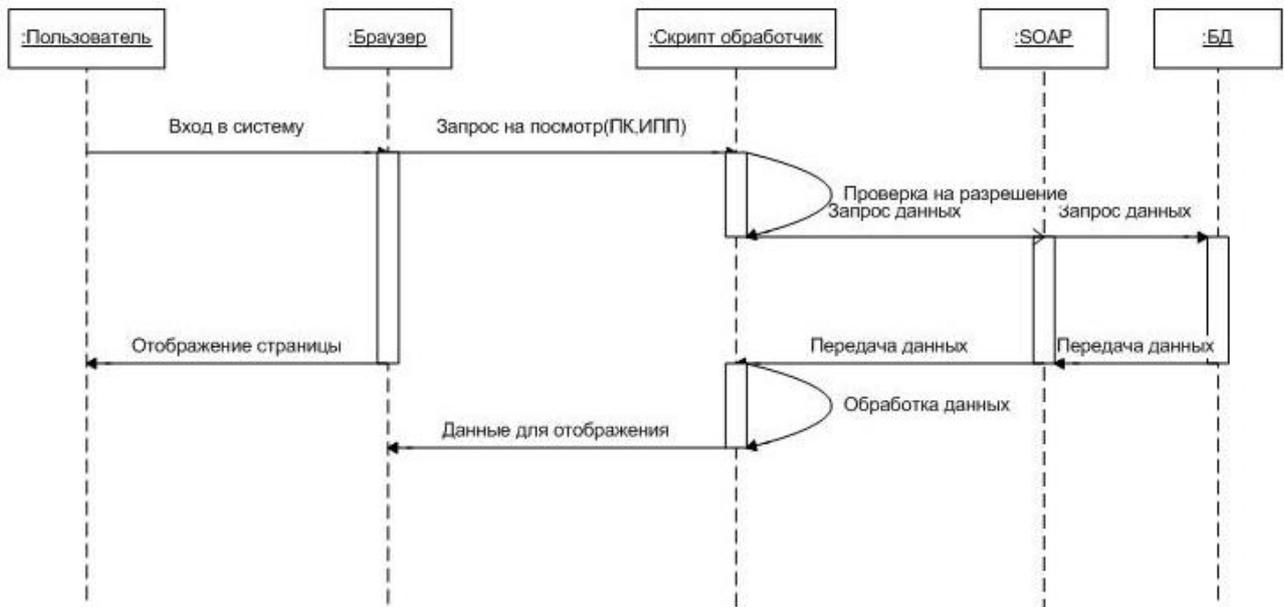
- Авторизация

Авторизация



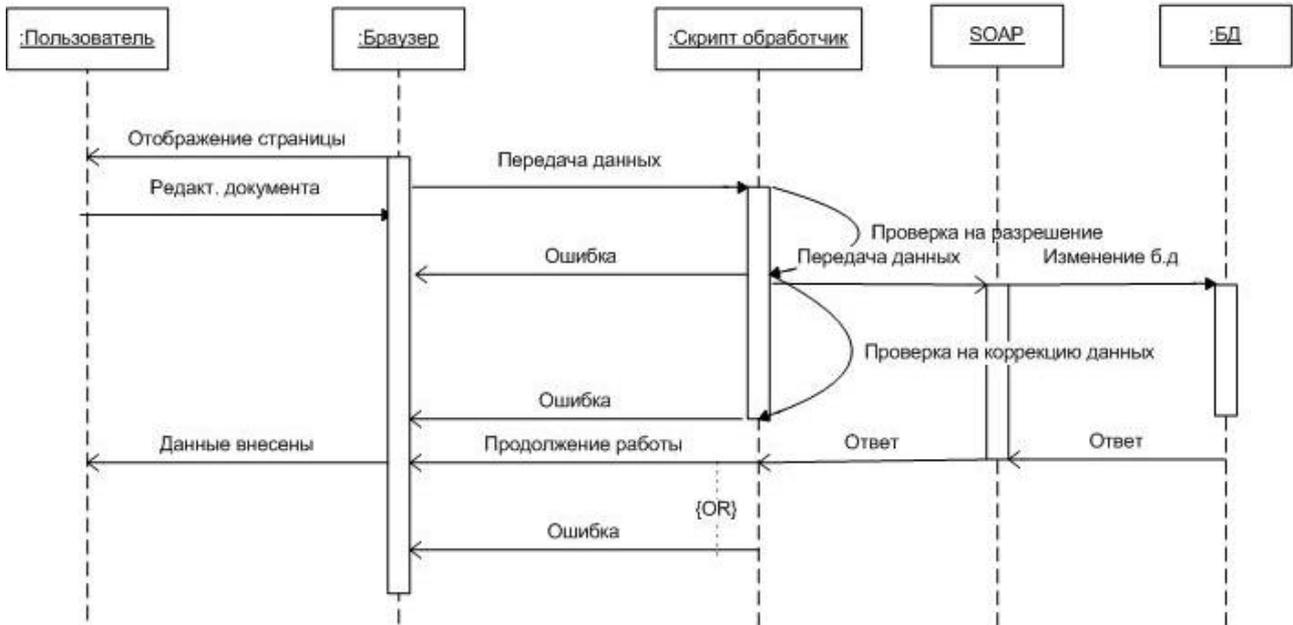
- Просмотр

Просмотр плана кафедры, ИПП



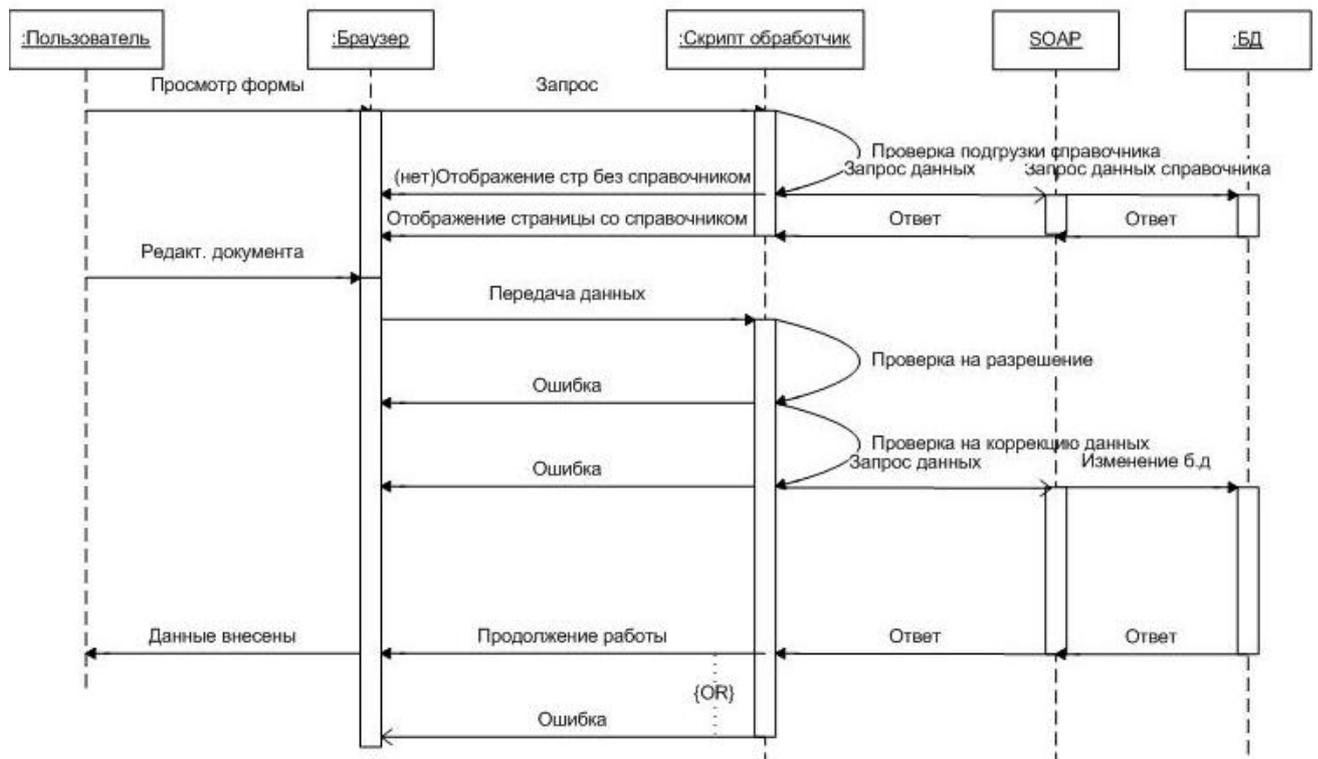
- Редактирование
Плана кафедры:

Редактирование плана кафедры



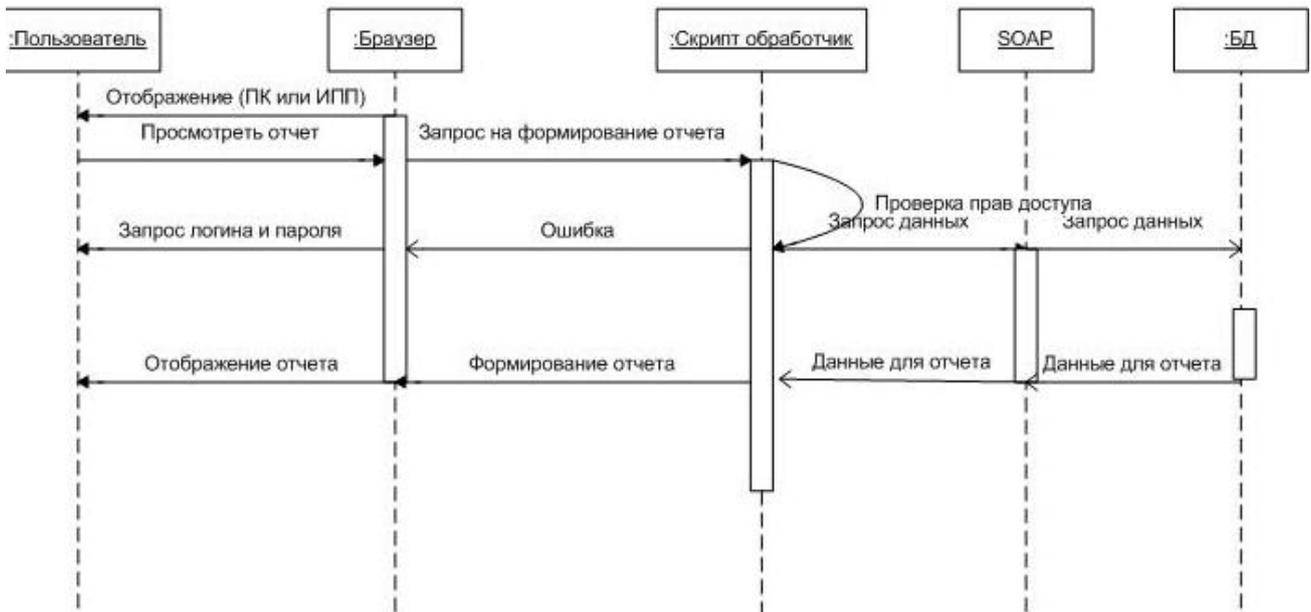
Индивидуального плана преподавателя:

Редактирование ИПП



- Формирование отчета

Формирование отчета



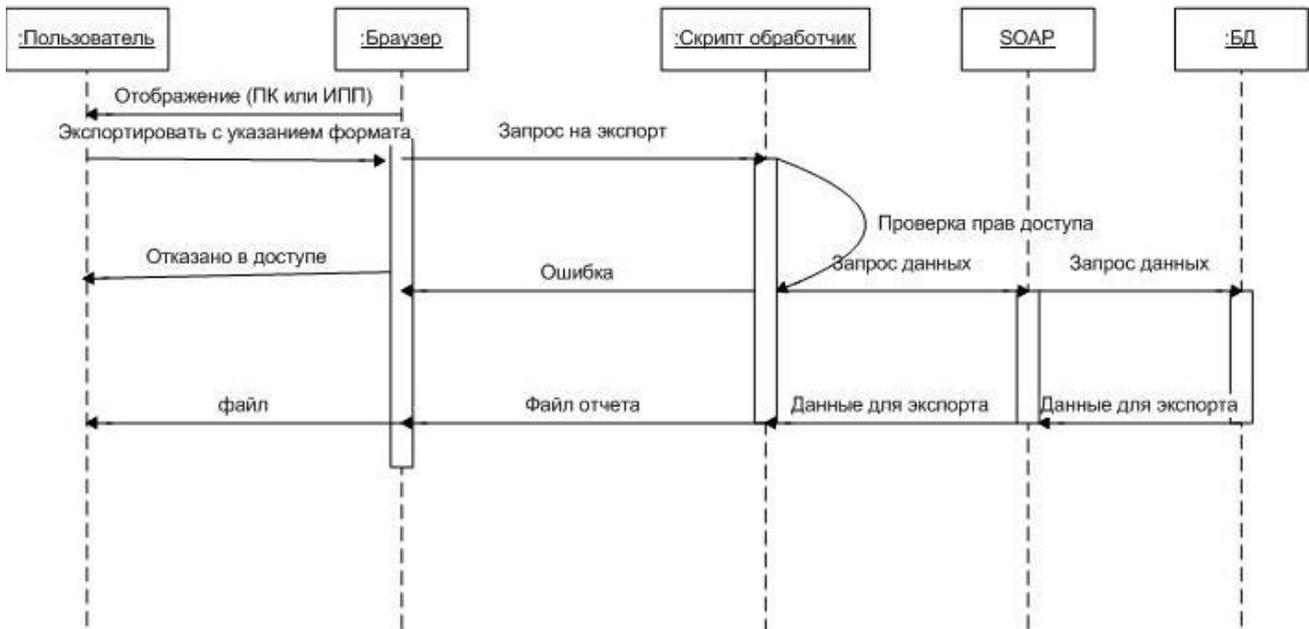
- Печать

Печать отчетов

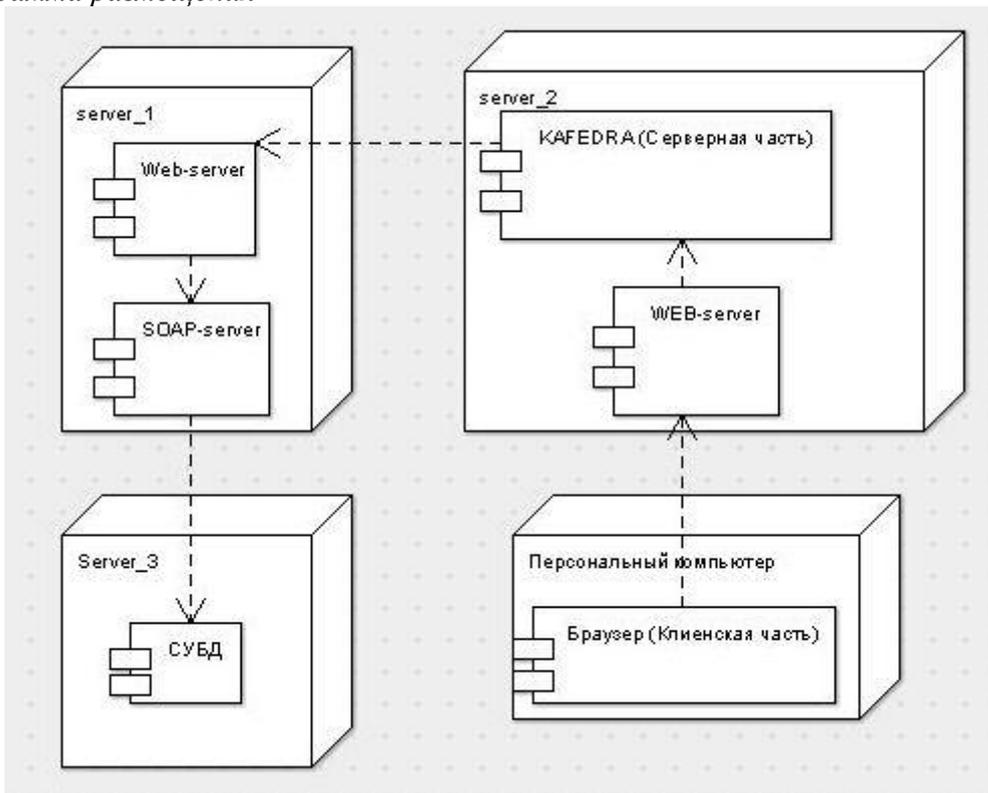


- Экспорт (документа в текстовый процессор)

Экспорт



3.2.2 Диаграмма размещения



3.3 Требования к производительности

Время отклика для типичных задач – не более 5 секунд.

3.5. Атрибуты системы

3.5.2. Доступность

Доступ к системе может быть получен через интернет-соединение с любого компьютера, имеющего HTML-совместимый браузер. Не требует установки на клиентскую машину. Клиентская часть обладает кросс-браузерной (для наиболее распространённых браузеров: MS IE, FireFox, Opera) и кросс-платформенной совместимостью.

3.5.3. Защита

В систему могут войти только те пользователи, у которых есть соответствующие права доступа. В системе предусмотрено несколько уровней прав доступа для разграничения полномочий пользователей. У каждого пользователя должен быть индивидуальный логин и пароль для входа в систему.

4. Дополнительная информация

4.2. Приложение

Расчет метрик (функциональных)

Информационные характеристики:

EI – количество внешних вводов,

EO – количество внешних выводов,

EIN – количество внешних запросов,

ILF – количество внутренних логических файлов,

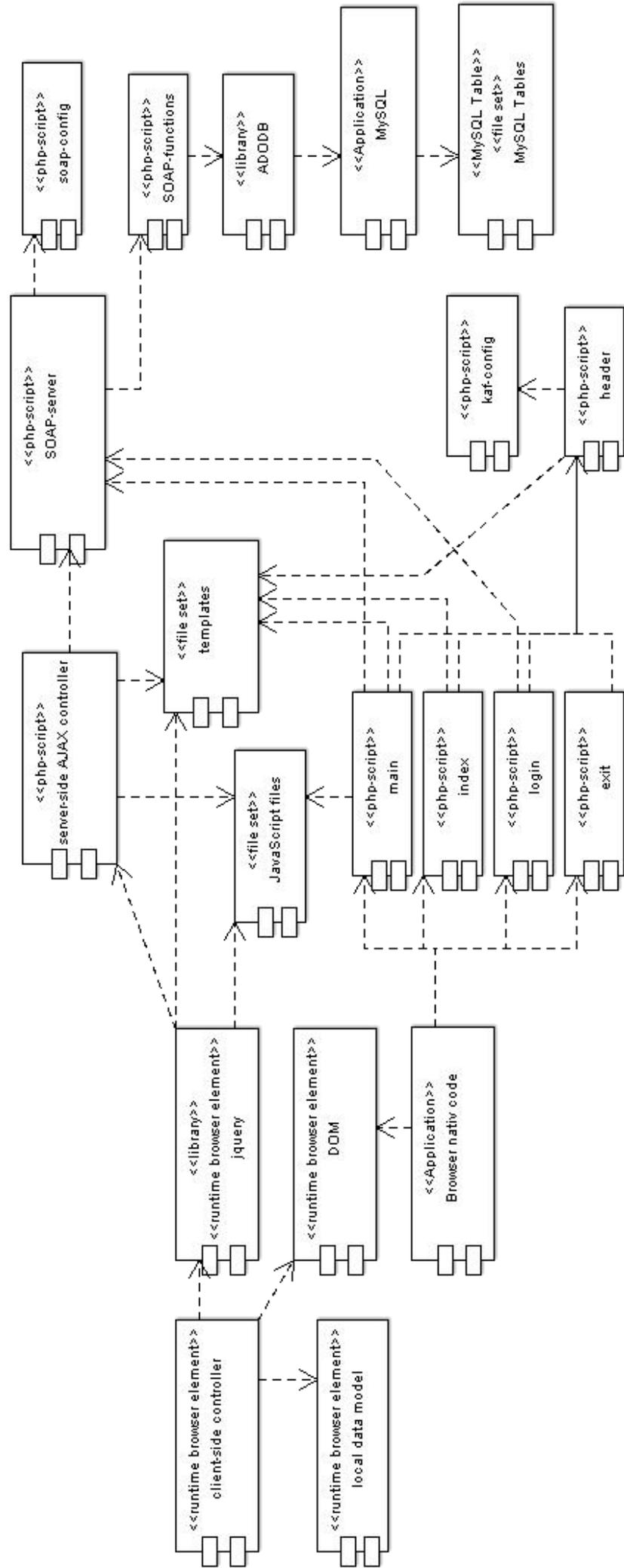
ELF – количество внешних интерфейсных файлов.

	EI	EO	EIN	ILF	ELF	Всего:
Авторизация	2	1	1	5	0	28
Просмотр	2	1	2	6	0	28
Редактирование	6	1	4	8	0	31
Формирование отчета	1	1	3	14	0	28
Экспорт	2	1	1	3	1	30
Печать	1	1	1	3	0	23
Итого:						168

Сумма коэффициентов регулировки сложности $\sum Fi=38$

Функциональные указатели $FP = 168 * (0.65 + 0.01 * 38) = 173.04$

Диаграмма развертывания



Задание 8

Задание выполняется в группе из 3-4 человек

Проектирование архитектуры приложения

1. Выбор языка программирования
2. Выбор фреймворка и/или библиотек для реализации интерфейса пользователя
3. Выбор СУБД
4. Разбить на подсистемы
5. Выделить компоненты
6. Нарисовать диаграмму компонентов

Для рисования диаграммы можно использовать

- <https://app.diagrams.net/> (aka draw.io)
- <https://www.umletino.com/umletino.html> (нет совместной работы)

Критерии оценки

- Выбран язык программирования - 1 балл
- Выбран фреймворк, библиотеки для реализации интерфейса пользователя - 2 балла
- Выбрана СУБД - 1 балл
- Сделано обоснование выбора - 1 балл
- Нарисована диаграмма компонентов (подсистем) - 1 балл
- Обозначения на диаграмме соответствуют стандарту UML - 2 балла
- Подсистемы и компоненты обеспечивают выполнение требований (функций) ПО - 2 балла

Итого - 10 баллов

Пример

1 Был выбран язык программирования C++, так как:

- C++ позволяет создавать кроссплатформенные приложения;
- Программы, созданные на C++ имеют низкие системные требования;
- Является одним из самых популярных языков, что положительно сказывается на поддержке ПО;

- Имеется библиотека для взаимодействия с MongoDB.

2 Для реализации интерфейса пользователя используется фреймворк Qt

- Один из самых популярных UI фреймворков;
- QtCreator позволяет создавать интерфейс в графическом режиме;
- Имеет большое сообщество и множество готовых библиотек.

3. В качестве СУБД была выбрана MongoDB так как:

- Позволяет создавать коллекции документов с гибкой структурой;
- Поддерживает массивы, бинарные данные, строки, числа и другие типы данных;
- Реализует принципы ACID;
- Поддержка многопоточности;
- Поддерживает потоки событий об изменении данных в базе данных;
- Близка к представлению данных в реальном мире.

4 Подсистемы:

1. Графический интерфейс

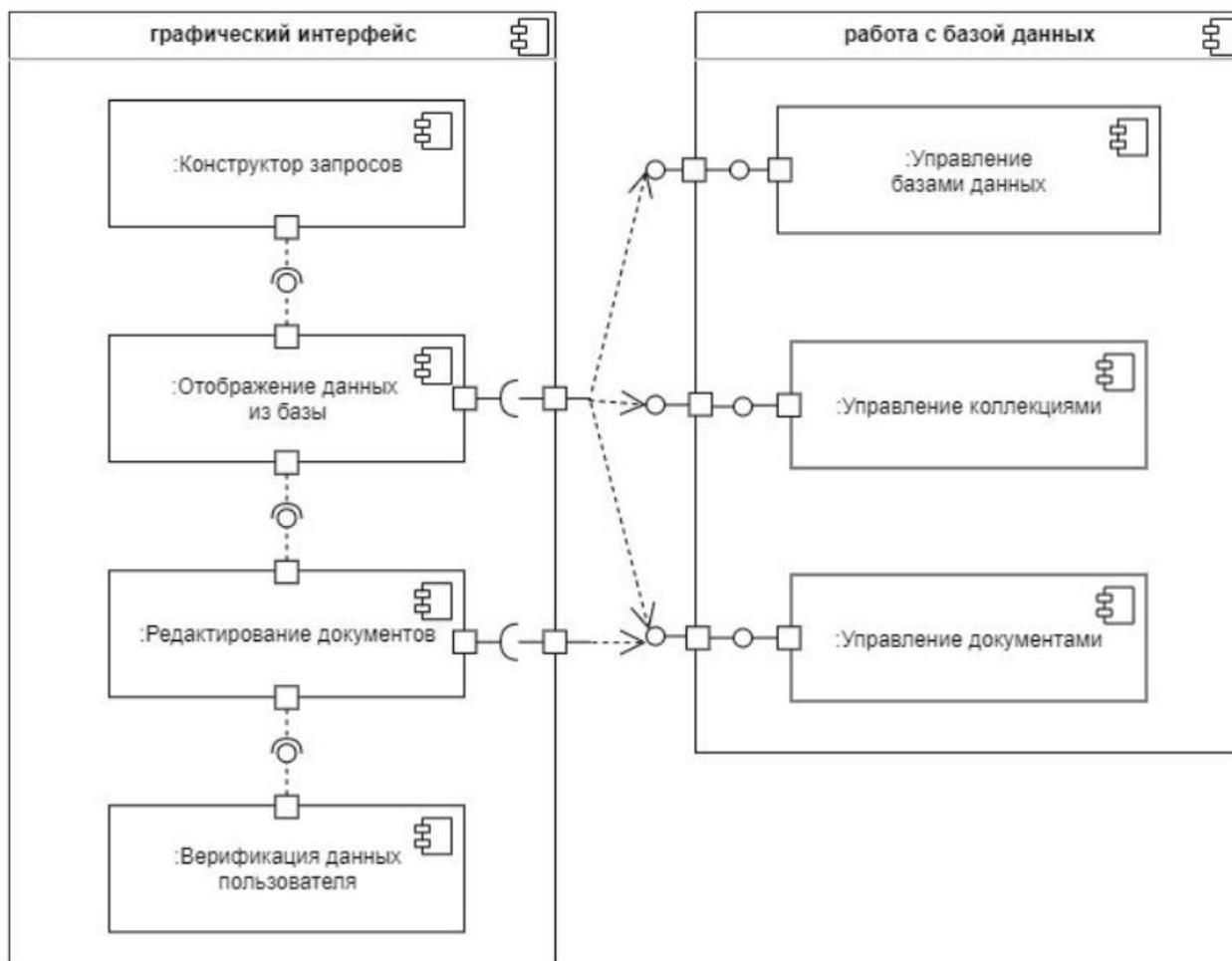
2. Работа с базой данных

5. Выделить компоненты

- Графический интерфейс: отображение данных из базы, конструктор запросов,

редактирование документов, верификация данных пользователя.

- Работа с базой данных: управление базами данных; управление коллекциями; управление документами.



Задание 9

Задание выполняется в группе из 3-4 человек

Разработка иерархии классов

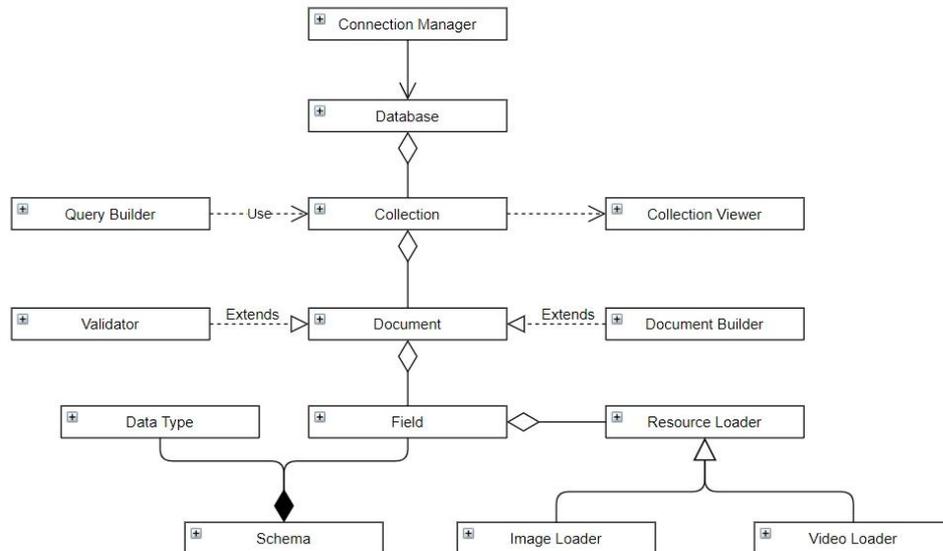
Разработка диаграммы классов проекта. Не делайте детализацию, только название классов.

Для выполнения задания можно использовать <https://app.diagrams.net/> (aka draw.io)

Критерии оценки

- Разработана диаграмма классов - 3 балла
- Правильно использованы обозначения UML - 3 балла
- Каждому компоненту (задание 5) соответствует хотя бы один класс - 2 балла
- Направление связей в иерархии классов обеспечивают устойчивость компонент - 2 балла

Пример



Задание 10

Детализированная диаграмма классов и спецификация модулей

Выбрать классы одной компоненты, нарисовать диаграмму классов для них.

Формат описания поля:

– имя : тип [кратность] = нач_значение {ограничения}

ограничения для полей: readOnly id , для полей с кратностью ordered unique nonunique
 подчеркивание имени static

Формат описания метода:

видимость имя(параметры):тип_результата {ограничения}

видимость (+ - #) может комбинироваться с / если метод переопределен

ограничения для методов: query - не изменяет состояния объекта курсив - чисто виртуальный метод

Для нескольких методов написать спецификацию (вход, выход, исключения)

Пример:

```
private ASection getCurrentSection(int startOffset, int endOffset)
Возвращает интервал в документе, выделенный в данный момент курсором,
или null если в документе сейчас нет активного выделения.
Параметры:
startOffset позиция начала выделения
endOffset позиция конца выделения
Исключения: нет
```

Критерии оценки

- Нарисована диаграмма классов (не менее 2 классов) - 3 балла
- Использованы корректные графические обозначения UML для классов и связей - 2 балла
- В диаграмме классов поля имеют правильные описания - 1 балл

- В диаграмме классов методы имеют правильные описания - 1 балл
- Указаны спецификации не менее 3 методов - 3 балла

Итого 10 баллов

Пример:

`public Document Collection::insert(Document document)` – добавляет документ “document” в коллекцию и возвращает тот же документ с установленным базой данных “id”.

Параметры:

document – документ для добавления в коллекцию

Исключения: `DocumentAlreadyExistsException`

`public void Document::set(string path, string value)` – устанавливает значение “value” по указанному пути “path” в документе.

path - разделяемый "." путь до поля. Например:

“user.address.city”

value - значение поля

Исключения: нет

`private void ConnectionManager::connect()` - устанавливает подключение к базе данных по заданным параметрам.

Исключения: `NoSuchHostException`, `AuthenticationFailedException`

`public vector<Database> ConnectionManager::list_databases()` - возвращает список баз данных, либо пустой список, если их нет.

Исключения: `NoConnectionException`



Задание 11

Диаграммы взаимодействия

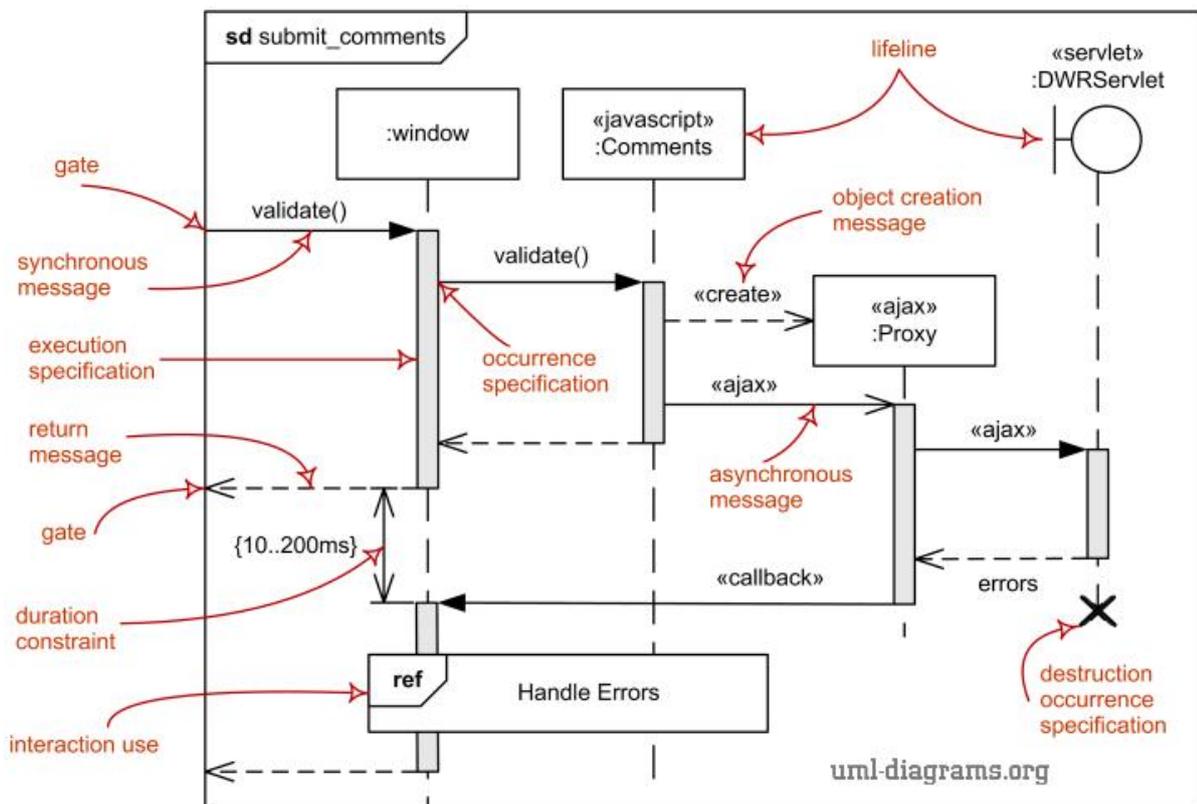
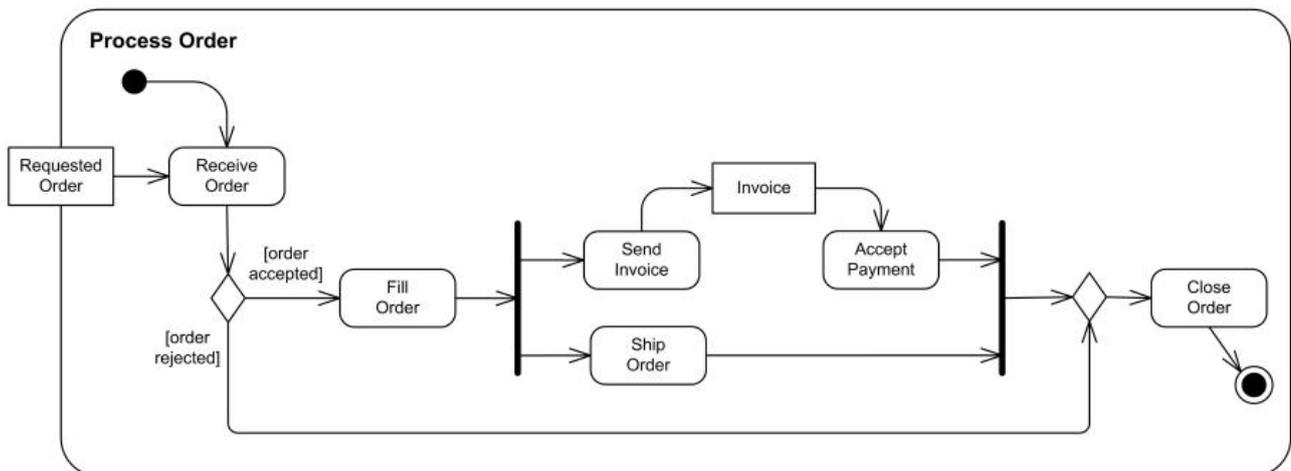
Для учебного проекта нарисовать диаграмму деятельности или диаграмму последовательности (процессы для диаграмм выбираются по согласованию с преподавателем)

Критерии оценивания

- Нарисована диаграмма - 3 балла
- Использованы правильные обозначения UML - 4 балла
- Соответствует выбранному процессу - 3 балла

Итого 10 баллов

Примеры



Задание 12

Детальное проектирование компонент интеллектуальных систем с использованием паттернов проектирования

Вариант 1

Выбор тарифного плана данного оператора сотовой связи

Вариант 2

Выбор специальности обучения в университете

Вариант 3

Выбор блюд в ресторане

Вариант 4

Рекомендательная система для фильмов

Критерии оценивания

Определен метод для получения результата — 2 балла, иначе 0 баллов

Определен способ интеграции компонента в систему— 2 балла, иначе 0 баллов

Интерфейс модулей соответствует задаче – 3 балла, оценка снижается на 1 балл за каждую ошибку в описании

Использованы паттерны проектирования – 3 балла (1 балл за каждый паттерн, но не более 3 баллов)

Итого 10 баллов

Вопросы к экзамену

Темы вопросов по разделу 1:

1. Определения: проблема, потребность, функция (feature), требования к ПО (функциональные, нефункциональные)
2. Анализ проблемы. Этапы анализа проблемы
3. Типы требований
4. Системный анализ при разработке ПО. Производные требования.
5. Преграды на пути выявления требований
6. Список способов выявления потребностей и требований (Лэффенгуэлл, Вигерс)
7. Интервьюирование (задачи, правила)
8. Совещание (задачи, правила)
9. Мозговой штурм (задачи, правила)
10. Раскадровки (задачи, правила)
11. Роль прототипирования в выявлении требований. Виды прототипирования ПО.
12. Диаграмма вариантов использования (задачи, обозначения)
13. Обыгрывание ролей (задачи, правила)

14. Выявление задач автоматизации с помощью методов искусственного интеллекта и машинного обучения. Ограничения методов ИИ и МО.
15. Оценивание пригодности исходных данных для экспертных систем и машинного обучения, эффекта от использования результатов
16. Документ-концепция, спецификация требований к ПО (назначение, рекомендации)
17. Способы описания требований, проверка качества описания на естественном языке.
18. Управление масштабом (задачи, методы)
19. Управление требованиями (задачи, методы)

Темы вопросов по разделу 2:

20. Диаграмма развертывания (назначение, обозначения)
21. Диаграмма классов (назначение, обозначения)
22. Диаграмма компонентов (назначение, обозначения)
23. Диаграмма объектов (назначение, обозначения)
24. Диаграмма пакетов (назначение, обозначения)
25. Диаграмма деятельности (назначение, обозначения)
26. Диаграмма состояний (назначение, обозначения)
27. Диаграмма последовательности (назначение, обозначения)
28. Диаграмма коммуникации (назначение, обозначения)
29. Влияние парадигм на архитектуру (ФП, ООП, СП)
30. Принципы SOLID
31. Принципы для связности компонентов (REP, CCP, CRP)
32. Принципы для сочетаемости компонентов (ADP, SDP, SAP)
33. Цель архитектуры, влияние на разработку, развертывание, эффективность работы, сопровождение.
34. Разделение архитектуры на горизонтальные и вертикальные уровни (используемые принципы). Границы системы, трансграничные вызовы и информация о структуре обрабатываемых данных.
35. Рекомендуемая "чистая архитектура" (Мартин), правило зависимостей.
36. Классификация методов искусственного интеллекта и машинного обучения.
37. Способы интеграция компонент ИИ в приложение.
38. Паттерны проектирования для интеллектуальных систем
39. CASE-средства для разработки интеллектуальных систем

01.03.02_Прикладная матем_ и искусств интеллект_Анализ требований и проектирование систем искусственного интеллекта_2024-2025_о

Проректор по учебной работе утверждено В.Е. Федоров

Ученым советом математического факультета

Протокол заседания № 3 от 19.10.2023

Председатель Ученого совета
математического факультета согласовано Е.А. Сбродова

Заседанием кафедры теории управления и оптимизации

Протокол заседания № 4 от 12.10.2023

Заведующий кафедрой согласовано И. В. Изместьев

Автор (составитель) Н.Д. Ливанов

Структура рабочей программы соответствует приказу ректора ФГБОУ ВО «ЧелГУ» от «13» апреля 2021 г. № 247-1