РЕФЕРАТ

Выпускная квалификационная работа: 62 страницы, 32 рисунка, 5 таблиц, 21 источник, 2 приложения.

Ключевые слова: web-приложение, корпоративная информационная система, Vue.js, micro-frontend, UML.

Объектом выпускной квалификационной работы является корпоративная информационная система Рубцовского института (филиала) АлтГУ.

Предметом выпускной квалификационной работы является процесс интеграции компонентов корпоративной информационной системы.

Цель выпускной квалификационной работы – разработка web-приложения интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ.

Для достижения поставленной цели необходимо выполнить следующие задачи:

– провести анализ предметной области;

– выявить недостатки существующей КИС Института;

– выработать проектные решения по функциональному обеспечению разрабатываемого web-приложения;

– описать проектные решения по обеспечивающим подсистемам разрабатываемого web-приложения;

– выполнить обзор программных продуктов;

– разработать web-приложение интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ;

– оценить эффективность от внедрения web-приложения.

Внедрение web-приложения позволит достичь следующих положительных сторон:

1. Единая точка идентификации и авторизации.
2. Представление информации при помощи web-браузера.
3. Возможность организации удаленной работы с корпоративной информационной системой Института.
4. Беспрерывная поставка обновлений ИС, а также удобное внедрение новых информационных систем в КИС.
5. Единая система оповещения и информирования сотрудников.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 4
1 Аналитическая часть 6
1.1 Технико-экономическая характеристика предметной области 6
1.2 Анализ функционирования объекта исследования 23
1.3 Определение цели и задач проектирования web-приложения 25
1.4 Обзор и анализ существующих разработок 26
1.5 Выбор и обоснование проектных решений 27
1.5.1 Выбор фреймворков, технологий разработки и технологических платформ 27
1.5.2 Выбор средств разработки и проектирования 29
2 Проектная часть 31
2.1 Разработка функционального обеспечения 31
2.2 Разработка информационного обеспечения 31
2.2.1 Используемые классификаторы и системы кодирования 31
2.2.2 Характеристика нормативно-справочной и входной оперативной информации 32
2.2.3 Характеристика результатной информации 32
2.3 Разработка программного обеспечения 34
2.3.1 Описание клиентских программных модулей 34
2.3.2 Компоненты пользовательского интерфейса 39
2.4 Компьютерно-сетевое обеспечение 41
2.5 Обеспечение информационной безопасности 42
3 Оценка эффективности внедрения информационной системы 47
3.1 Общие положения 47
3.2 Показатели эффективности 48
3.3 Расчет экономической эффективности 50
3.3.1 Смета затрат на разработку 50
3.3.2 Оценка экономической эффективности 55

ЗАКЛЮЧЕНИЕ 57
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 59
ПРИЛОЖЕНИЕ А 63
ПРИЛОЖЕНИЕ Б 64

ВВЕДЕНИЕ

На сегодняшний день в каждой организации функционируют различные информационные системы, в процессе развития и усложнения которых, предприятия выстраивают корпоративные информационные системы (КИС), работающие в комплексе друг с другом.

Корпоративная информационная система строится на основе единого информационного пространства, охватывая и координируя всю совокупность управленческих процессов предприятия.

В рамках развития корпоративной информационной системы Рубцовского Института (филиала) АлтГУ, перед сотрудниками отдела технического и программного обеспечения стоит задача по усовершенствованию информационных систем для сотрудников и студентов Института.

Для реализации данной задачи необходимо осуществить перенос всех действующих ИС в веб-приложения, что позволит упростить обслуживание, настройку, а также эксплуатацию рабочих компьютеров сотрудников, организовать возможность удаленного доступа к локальным ИС и ресурсам.

Так как в Рубцовском институте (филиале) АлтГУ для работы сотрудников существует большое количество программных продуктов, необходимо, организовать единое информационное пространство – «Единое окно», с помощью которого и будет осуществляться доступ к ресурсам Института.

В связи с этим, объектом выпускной квалификационной работы является корпоративная информационная система Рубцовского института (филиала) АлтГУ.

Предметом выпускной квалификационной работы является процесс интеграции компонентов корпоративной информационной системы.

Цель выпускной квалификационной работы – разработка web-приложения интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ.

Для достижения поставленной цели необходимо выполнить следующие задачи:

– провести анализ предметной области;

– выявить недостатки существующей КИС Института;

– выработать проектные решения по функциональному обеспечению разрабатываемого web-приложения;

– описать проектные решения по обеспечивающим подсистемам разрабатываемого web-приложения;

– выполнить обзор программных продуктов;

– разработать web-приложение интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ;

– оценить эффективность от внедрения web-приложения.

Методы, используемые при написании работы: системный анализ, моделирование предметной области.

Ресурсами информации, используемой в данной работе, являются Интернет, учебно-методическая и научная литература, локальная нормативно-справочная документация Рубцовского института (филиала) АлтГУ.

При проектировании информационной системы использовались:

– UML Designer – графический инструмент для редактирования и визуализации моделей UML;

– Vue.js – JavaScript-фреймворк с открытым исходным кодом для создания пользовательских интерфейсов;

– WebStorm – интегрированная среда разработки на JavaScript.

1 Аналитическая часть

1.1 Технико-экономическая характеристика предметной области

Рубцовский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Алтайский государственный университет» (далее – Институт) – обособленное структурное подразделение ФГБОУ ВО «Алтайский Государственный университет» (далее – Университет), расположенное в городе Рубцовск Алтайского края и осуществляющее постоянно его функции в рамках выданной институту лицензии на образовательную деятельность [1].

Институт является государственной, некоммерческой организацией, имеет право на ведение образовательной деятельности и на льготы, установленные законодательством РФ, в соответствии с лицензией, выданной Министерством науки и высшего образования РФ [1].

Институт имеет полное официальное наименование: Рубцовский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Алтайский государственный университет». Сокращенное наименование Института: Рубцовский институт (филиал) АлтГУ.

Основными видами деятельности Института являются:

1) образовательная деятельность по образовательным программам высшего образования и среднего профессионального образования, основным программам профессионального обучения, а также дополнительным общеобразовательным программам и дополнительным профессиональным программам;

2) научная деятельность;

3) организация проведения социально значимых мероприятий в сфере образования и науки.

К компетенции Института относится:

1) организация и методическое обеспечение образовательного процесса;

2) подбор и расстановка педагогических и научных кадров, ответственность за уровень их квалификации;

3) формирование структура Института, создание факультетов, отделений, кафедр, центров, подготовительных отделений и курсов, научно-исследовательских лабораторий и иных подразделений;

4) формирование контингента обучающихся и выполнение контрольных цифр приёма;

5) осуществление текущего контроля успеваемости и промежуточной аттестации обучающихся в соответствии с Уставом Университета и Положением о Рубцовском институте (филиале) АлтГУ;

6) улучшение материально-технического обеспечения и оснащения образовательного процесса, оборудование помещений в соответствии с государственными и местными нормами и требованиями;

7) создание временных творческих коллективов с привлечением преподавателей других высших учебных заведений, ведущих ученых, специалистов и практиков на условиях совместительства, почасовой оплаты, договоров подряда для проведения научно-исследовательских разработок и их внедрения по заказам предприятий, организаций и органов местного самоуправления;

8) осуществление иной деятельности, не запрещенной законодательством Российской Федерации и предусмотренной Уставом Университета и Положением о Рубцовском институте (филиале) АлтГУ.

Основными функциями Института являются:

– реализация образовательных программ в сфере высшего профессионального образования по очной, заочной и очно-заочной

(вечерней) формам обучения по направлениям подготовки (специальностям) Алтайского Государственного Университета в соответствии с лицензией, выданной институту;

– организация учебного процесса в институте по образовательным программам высшего профессионального образования регламентируется учебным планом по направлению подготовки специальности и расписанием учебных занятий для каждой формы обучения, которые разрабатываются институтом самостоятельно и утверждаются на основе государственного образовательного стандарта высшего профессионального образования, примерных образовательных программ, учебных планов по направлению подготовки специальности и программам дисциплин, утверждаемых федеральным органом управления образованием, при этом примерный учебный план и программы дисциплин имеют рекомендательный характер;

– реализация программ дополнительного образования (по полной и ускоренной программам) на базе очного или заочного обучения;

– осуществление гибких программ для курсов переподготовки и повышения квалификации работников предприятий, организаций, учреждений и органов административного управления городов и поселков юга Алтайского края;

– разработка и внедрение инновационных технологий обучения;

– исследования в области предпринимательской деятельности, экономики, региональных и муниципальных реформ.

Организационная структура – совокупность звеньев, расположенных в строгой соподчиненности и обеспечивающих взаимосвязь между другими подразделениями организации, а также распределение между ними ответственности и прав, которая проявляется через разделение труда, создание специализированных подразделений, иерархию должностей и является необходимым элементом эффективной организации, так как

придает ей внутреннюю стабильность и позволяет добиться определенного порядка в использовании ресурсов.

Организационная структура Рубцовского института (филиала) АлтГУ представлена на рисунке 1.1.

Институт располагается в двух учебных корпусах. Численный состав сотрудников филиала составляет 147 человек. В филиале обучается более 1400 студентов, по различным формам обучения.

Для общего руководства Институтом создается коллегиальный выборный представительный орган – Учёный совет Института.

Непосредственное управление и текущее руководство деятельностью Института осуществляет директор.

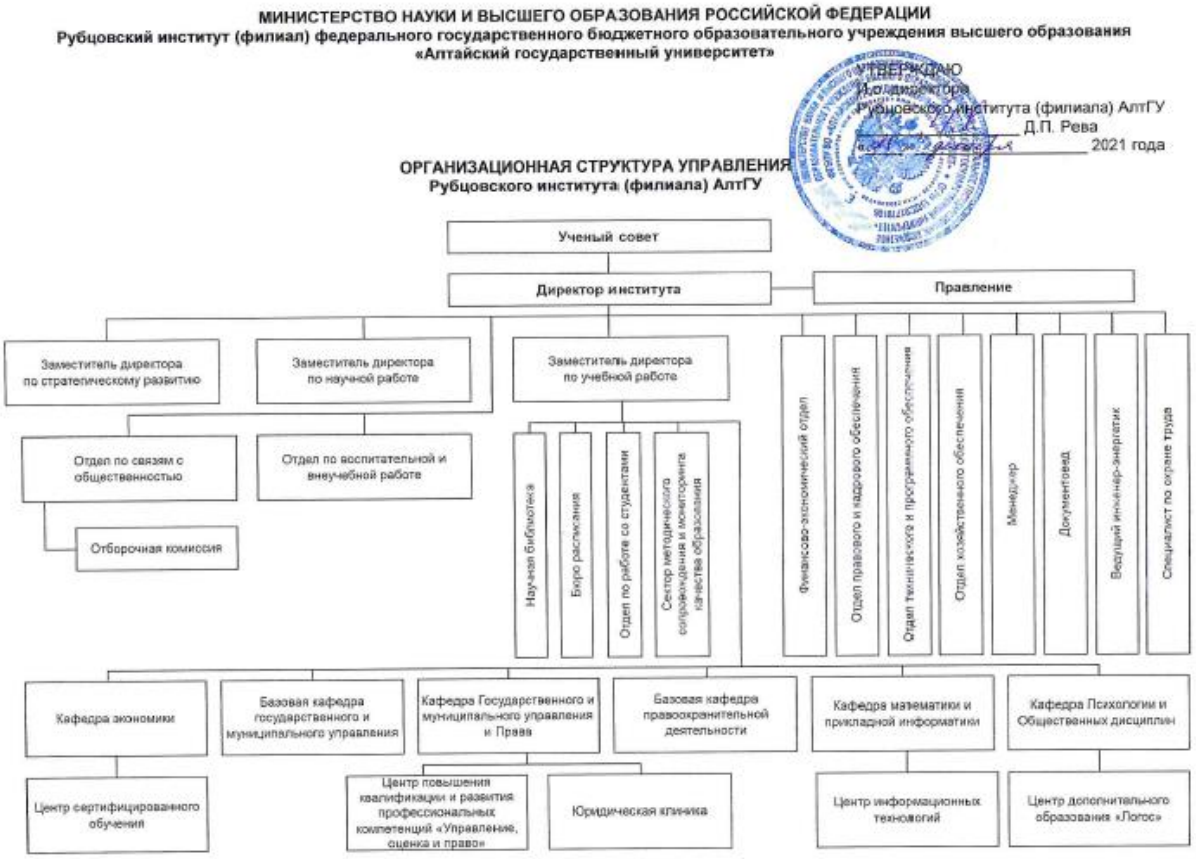


Рисунок 1.1 – Организационная структура управления Рубцовского института (филиала) АлтГУ

Каждое структурное подразделение в Институте имеет свою функциональную особенность. В целях облегчения и ускорения

взаимодействия структурных подразделений по решению ежедневных рабочих вопросов в организации существует электронный документооборот, при котором делопроизводство осуществляется с помощью высокоскоростной локальной сети.

Управление учебным процессом Института осуществляется за счет:

1. Информационных систем:

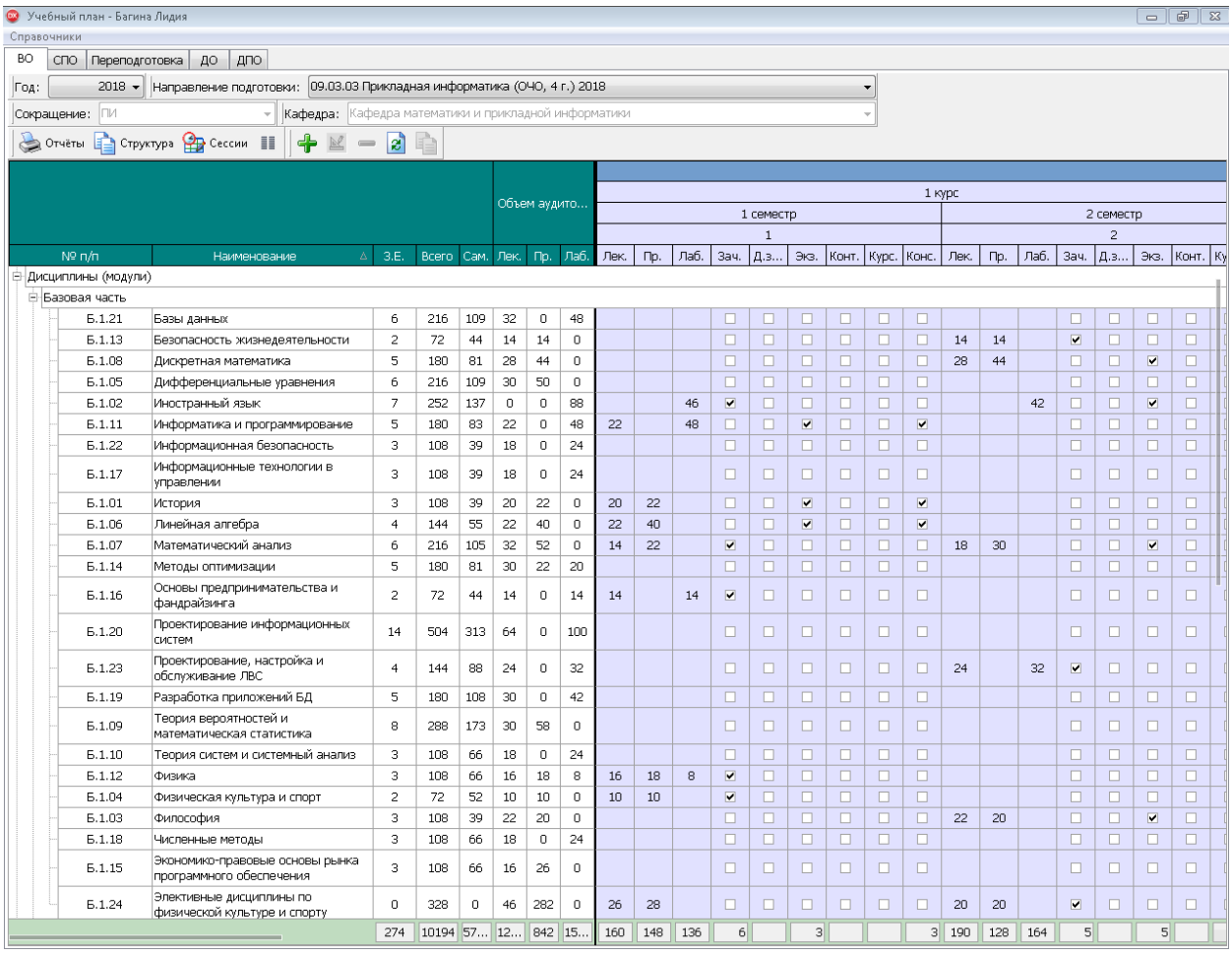
– «Учебные планы»;
– «Графики учебного процесса»;
– «Поручения»;
– «Компетенции»;
– «РПД/ФОС»;
– «Редактор расписания занятий»;
– «Абитуриенты»;
– «Студенты»;
– «Кадры».

2. Подсистем:

– «Электронная научная библиотека института»;
– «Система электронного обучения на платформе Moodle»;
– «Информационно-справочное окно».

Вышеперечисленные информационные системы являются разработкой Института и отвечают всем предъявляемым в настоящее время требованиям к подобным ИС.

ИС «Учебные планы» – система, в которой на каждый учебный год создаются или обновляются рабочие учебные планы направлений подготовки высшего образования и специальностей среднего профессионального образования в соответствии с ФГОС, а также учебные планы дополнительного профессионального образования (ДПО), переподготовок, дополнительного образования детей и взрослых (ДО) (рисунок 1.2).



№ п/п	Наименование	З.Е.	Всего	Сам.	Объем аудито... Лек.	Пр.	Лаб.	1 курс																		
								1 семестр									2 семестр									
								1									2									
								Лек.	Пр.	Лаб.	Зач.	Д.з...	Экз.	Конт.	Курс.	Конс.	Лек.	Пр.	Лаб.	Зач.	Д.з...	Экз.	Конт.	Ку
Дисциплины (модули)																										
Базовая часть																										
Б.1.21	Базы данных	6	216	109	32	0	48				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.13	Безопасность жизнедеятельности	2	72	44	14	14	0				☐	☐	☐	☐	☐	☐	14	14		☑	☐	☐	☐			
Б.1.08	Дискретная математика	5	180	81	28	44	0				☐	☐	☐	☐	☐	☐	28	44		☐	☐	☑	☐			
Б.1.05	Дифференциальные уравнения	6	216	109	30	50	0				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.02	Иностранный язык	7	252	137	0	0	88			46	☑	☐	☐	☐	☐	☐			42	☐	☐	☑	☐			
Б.1.11	Информатика и программирование	5	180	83	22	0	48	22		48	☐	☐	☑	☐	☐	☑				☐	☐	☐	☐			
Б.1.22	Информационная безопасность	3	108	39	18	0	24				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.17	Информационные технологии в управлении	3	108	39	18	0	24				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.01	История	3	108	39	20	22	0	20	22		☐	☐	☑	☐	☐	☑				☐	☐	☐	☐			
Б.1.06	Линейная алгебра	4	144	55	22	40	0	22	40		☐	☐	☑	☐	☐	☑				☐	☐	☐	☐			
Б.1.07	Математический анализ	6	216	105	32	52	0	14	22		☑	☐	☐	☐	☐	☐	18	30		☐	☐	☑	☐			
Б.1.14	Методы оптимизации	5	180	81	30	22	20				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.16	Основы предпринимательства и фандрайзинга	2	72	44	14	0	14	14		14	☑	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.20	Проектирование информационных систем	14	504	313	64	0	100				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.23	Проектирование, настройка и обслуживание ЛВС	4	144	88	24	0	32				☐	☐	☐	☐	☐	☐	24		32	☑	☐	☐	☐			
Б.1.19	Разработка приложений БД	5	180	108	30	0	42				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.09	Теория вероятностей и математическая статистика	8	288	173	30	58	0				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.10	Теория систем и системный анализ	3	108	66	18	0	24				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.12	Физика	3	108	66	16	18	8	16	18	8	☑	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.04	Физическая культура и спорт	2	72	52	10	10	0	10	10		☑	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.03	Философия	3	108	39	22	20	0				☐	☐	☐	☐	☐	☐	22	20		☐	☐	☑	☐			
Б.1.18	Численные методы	3	108	66	18	0	24				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.15	Экономико-правовые основы рынка программного обеспечения	3	108	66	16	26	0				☐	☐	☐	☐	☐	☐				☐	☐	☐	☐			
Б.1.24	Элективные дисциплины по физической культуре и спорту	0	328	0	46	282	0	26	28		☐	☐	☐	☐	☐	☐	20	20		☑	☐	☐	☐			
		274	10194	57...	12...	842	15...	160	148	136	6		3			3	190	128	164	5		5				

Рисунок 1.2 – Информационная система «Учебные планы»

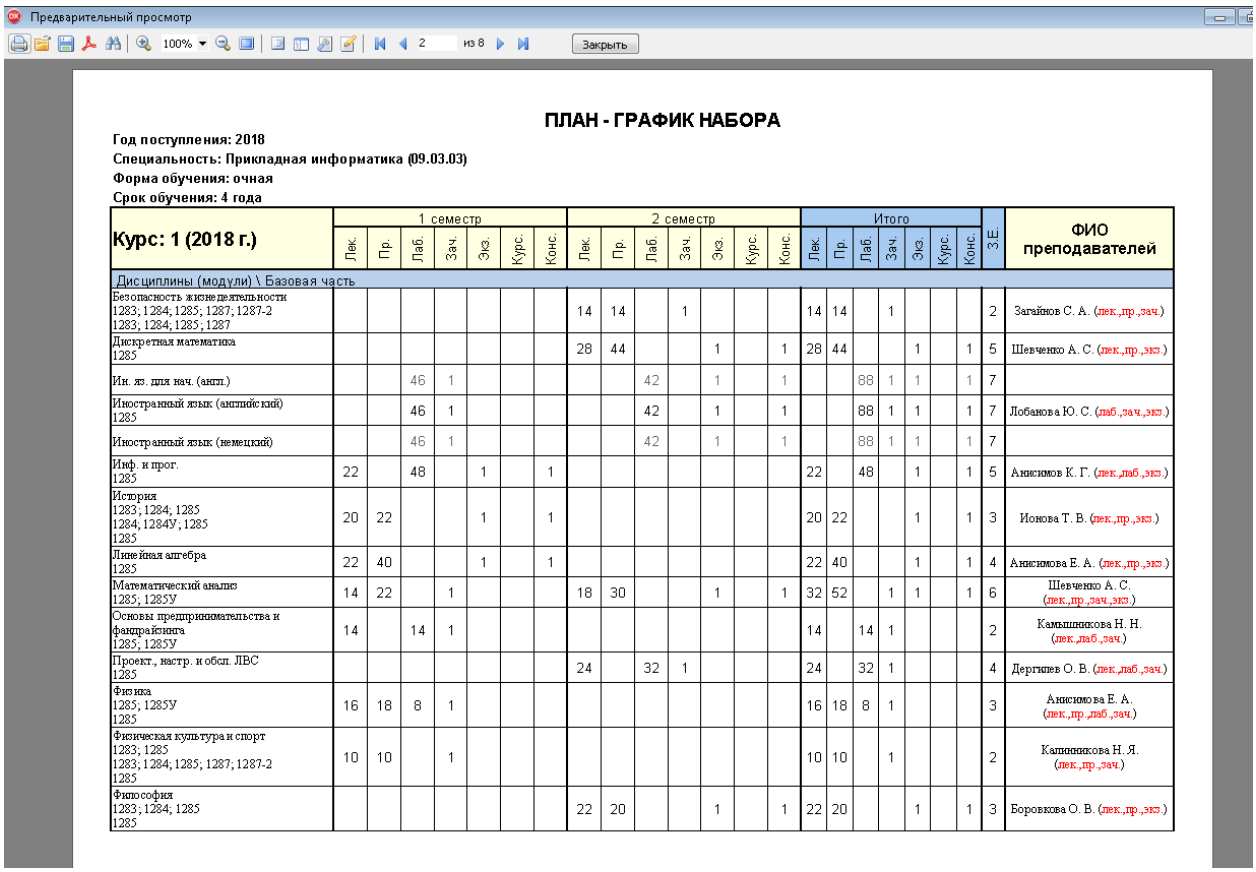
После внесения учебного плана специалистами по учебно-методической работе кафедр в данную информационную систему, появляется возможность формирования отчетов:

– «Учебный план»;

– «План график специальности»;

– «План-график набора» по каждому курсу, который выдается лично на руки каждому студенту (рисунок 1.3).

Учебные планы направлений подготовки ВО и специальностей СПО формируются один раз в год, а учебные планы ДПО, ДО, переподготовок обновляются в течение всего года по мере комплектования групп или индивидуального обучения (рисунок 1.4).



Предварительный просмотр

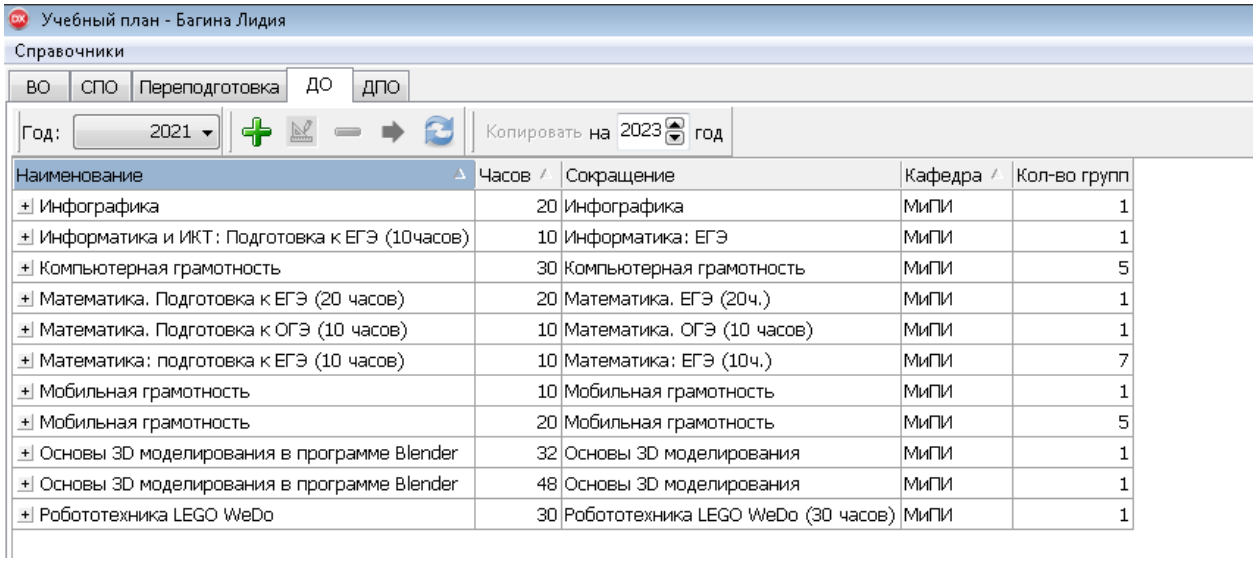
Закрыть

ПЛАН - ГРАФИК НАБОРА

Год поступления: 2018
Специальность: Прикладная информатика (09.03.03)
Форма обучения: очная
Срок обучения: 4 года

Курс: 1 (2018 г.)	1 семестр							2 семестр							Итого							З.Е.	ФИО преподавателей
	Лек.	Пр.	Лаб.	Зач.	Экз.	Курс.	Конс.	Лек.	Пр.	Лаб.	Зач.	Экз.	Курс.	Конс.	Лек.	Пр.	Лаб.	Зач.	Экз.	Курс.	Конс.		
Дисциплины (модули) \ Базовая часть																							
Безопасность жизнедеятельности 1283; 1284; 1285; 1287; 1287-2 1283; 1284; 1285; 1287								14	14		1				14	14		1				2	Загайнов С. А. (лек.,пр.,зач.)
Дискретная математика 1285								28	44			1		1	28	44			1		1	5	Шевченко А. С. (лек.,пр.,экз.)
Ин. яз. для нач. (англ.)			46	1						42		1		1			88	1	1		1	7	
Иностранный язык (английский) 1285			46	1						42		1		1			88	1	1		1	7	Лобанова Ю. С. (лаб.,зач.,экз.)
Иностранный язык (немецкий)			46	1						42		1		1			88	1	1		1	7	
Инф. и прог. 1285	22		48		1		1								22		48		1		1	5	Анисимов К. Г. (лек.,лаб.,экз.)
История 1283; 1284; 1285 1284; 1284У; 1285 1285	20	22			1		1								20	22			1		1	3	Ионова Т. В. (лек.,пр.,экз.)
Линейная алгебра 1285	22	40			1		1								22	40			1		1	4	Анисимова Е. А. (лек.,пр.,экз.)
Математический анализ 1285; 1285У	14	22		1				18	30			1		1	32	52		1	1		1	6	Шевченко А. С. (лек.,пр.,зач.,экз.)
Основы предпринимательства и фандрайзинга 1285; 1285У	14		14	1											14		14	1				2	Камышникова Н. Н. (лек.,лаб.,зач.)
Проект., настр. и обсл. ЛВС 1285								24		32	1				24		32	1				4	Дергилев О. В. (лек.,лаб.,зач.)
Физика 1285; 1285У 1285	16	18	8	1											16	18	8	1				3	Анисимова Е. А. (лек.,пр.,лаб.,зач.)
Физическая культура и спорт 1283; 1285 1283; 1284; 1285; 1287; 1287-2 1285	10	10		1											10	10		1				2	Калинникова Н. Я. (лек.,пр.,зач.)
Философия 1283; 1284; 1285 1285								22	20			1		1	22	20			1		1	3	Боровкова О. В. (лек.,пр.,экз.)

Рисунок 1.3 – Отчет «План-график набора»



Учебный план - Багина Лидия

Справочники

ВО | СПО | Переподготовка | ДО | ДПО

Год: 2021 | Копировать на 2023 год

Наименование	Часов	Сокращение	Кафедра	Кол-во групп
Инфографика	20	Инфографика	МиПИ	1
Информатика и ИКТ: Подготовка к ЕГЭ (10часов)	10	Информатика: ЕГЭ	МиПИ	1
Компьютерная грамотность	30	Компьютерная грамотность	МиПИ	5
Математика. Подготовка к ЕГЭ (20 часов)	20	Математика. ЕГЭ (20ч.)	МиПИ	1
Математика. Подготовка к ОГЭ (10 часов)	10	Математика. ОГЭ (10 часов)	МиПИ	1
Математика: подготовка к ЕГЭ (10 часов)	10	Математика: ЕГЭ (10ч.)	МиПИ	7
Мобильная грамотность	10	Мобильная грамотность	МиПИ	1
Мобильная грамотность	20	Мобильная грамотность	МиПИ	5
Основы 3D моделирования в программе Blender	32	Основы 3D моделирования	МиПИ	1
Основы 3D моделирования в программе Blender	48	Основы 3D моделирования	МиПИ	1
Робототехника LEGO WeDo	30	Робототехника LEGO WeDo (30 часов)	МиПИ	1

Рисунок 1.4 – Вкладка «ДО» ИС «Учебный план»

В информационную систему «График учебного процесса» вносятся графики, утвержденные Ученым советом Института (рисунок 1.5).

В графике отражаются направления подготовки ВО и специальности

СПО с указанием количества лет обучения, а также номера групп. Для каждой группы указывается период аудиторного обучения, практик, сессий и каникул.

Информационная система позволяет экспортировать график учебного процесса в файл Microsoft Excel.

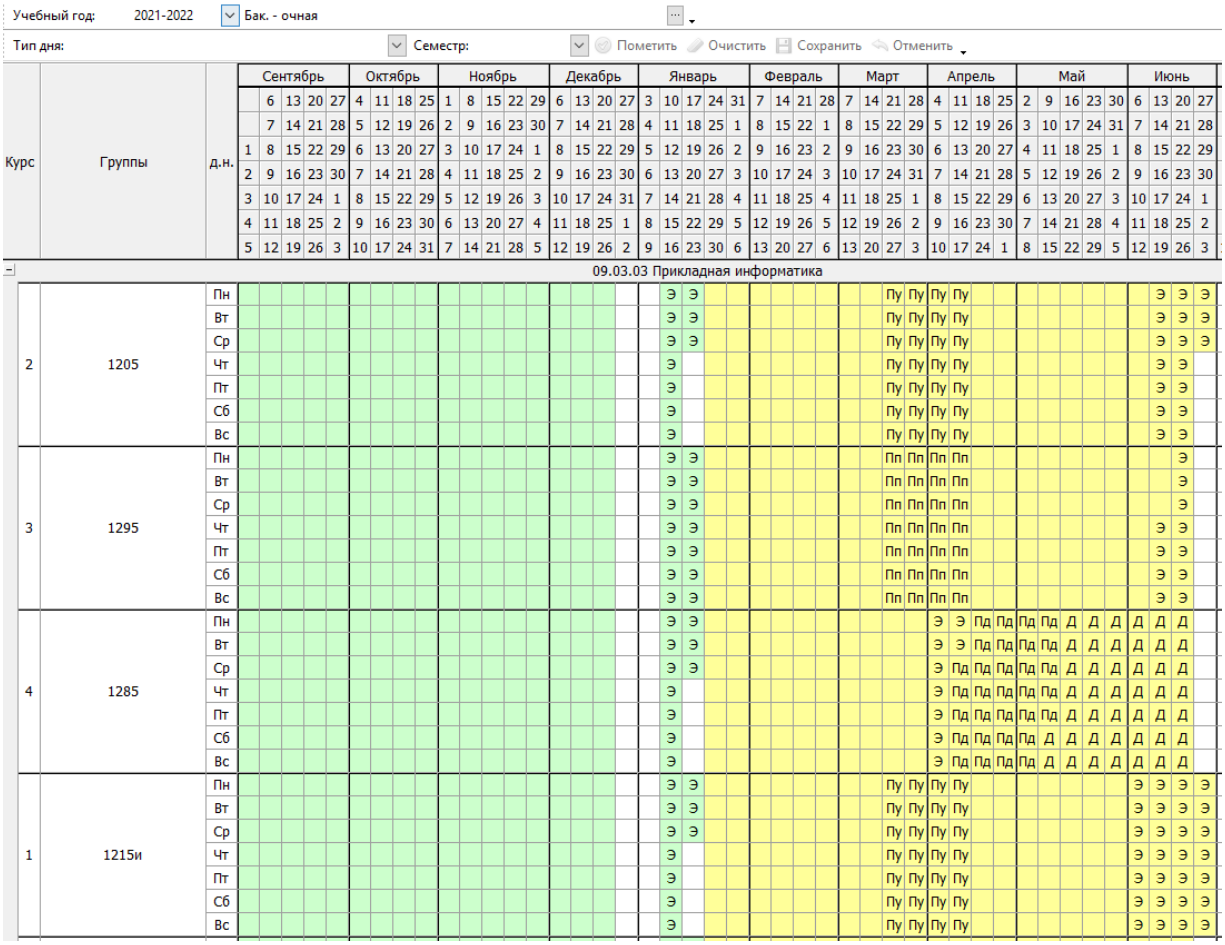


Рисунок 1.5 – Информационная система «График учебного процесса»

В ИС «Поручения» на основании сформированных учебных планов, количества студентов в группах и информации из ИС «Кадры» (должность преподавателя, количество часов на ставку и т.д.) (рисунок 1.6), формируется учебная нагрузка преподавателей на учебный год, имеется возможность закреплять каждого преподавателя отдельно за видами занятий: лекциями, практическими занятиями, лабораторными занятиями и промежуточной аттестацией в семестре (рисунок 1.7).

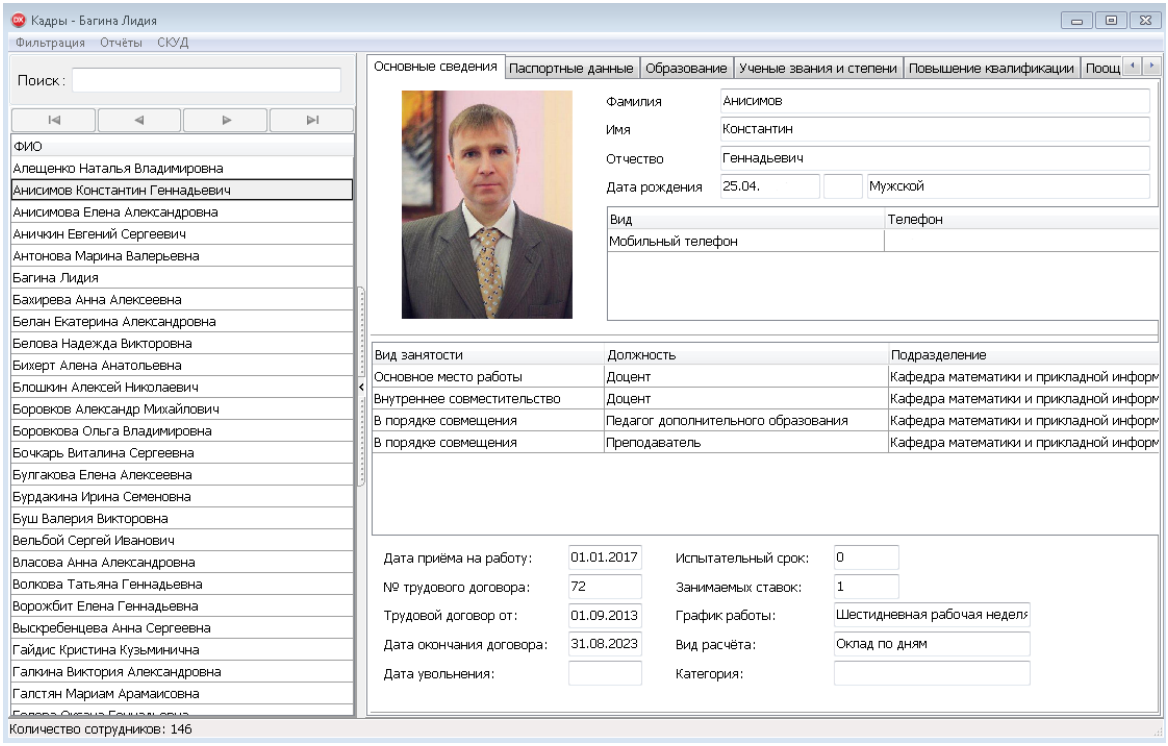


Рисунок 1.6 – Информационная система «Кадры»

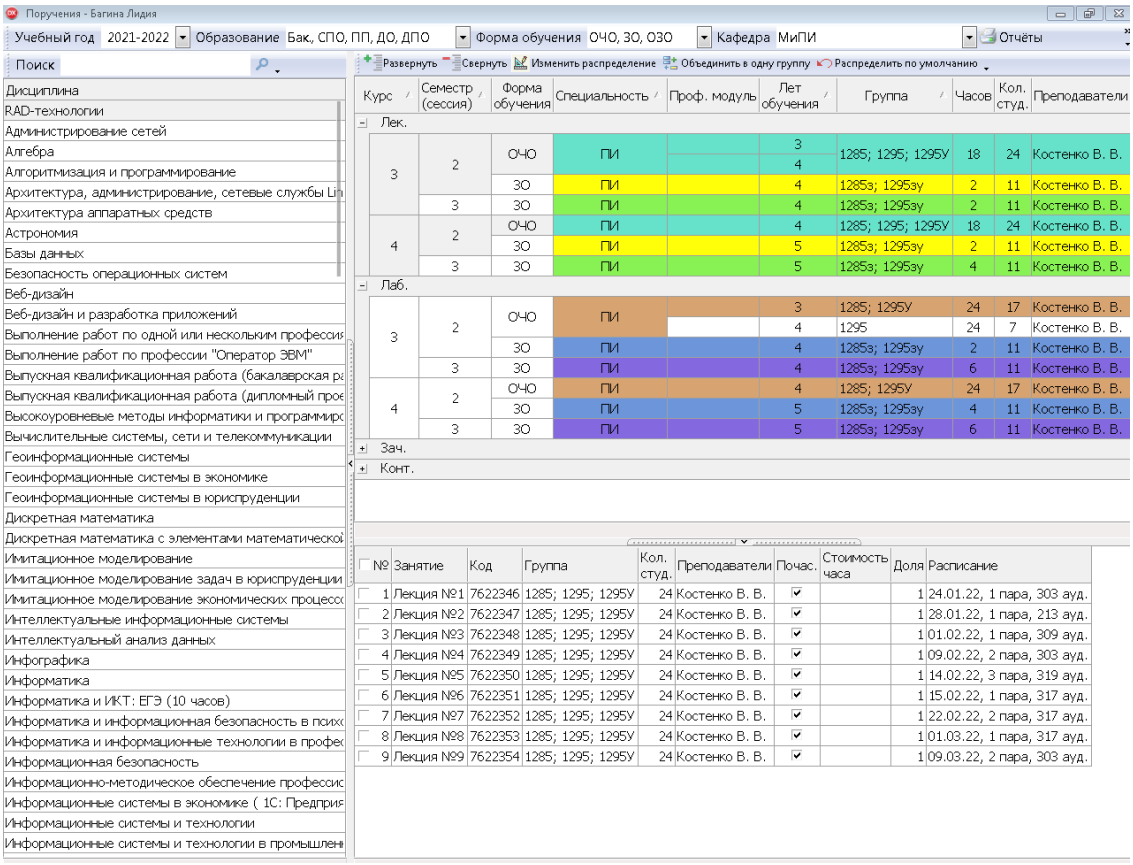


Рисунок 1.7 – Информационная система «Поручения»

ИС «Поручения» позволяет формировать множество отчетов:

1. Учебное поручение с возможностью выбора года распределения (на каждого преподавателя кафедры или общее на всех преподавателей кафедры) (рисунок 1.8):

– подробное;

– итоги;

– по видам занятий;

– по семестрам;

– по дисциплинам.

2. Нагрузка:

– по кафедре (рисунок 1.9);

– нераспределенная.

3. Отчеты о выполнении почасовой нагрузки:

– автозаполненный отчёт;

– пустой отчёт.

4. Распределение ставок ППС.

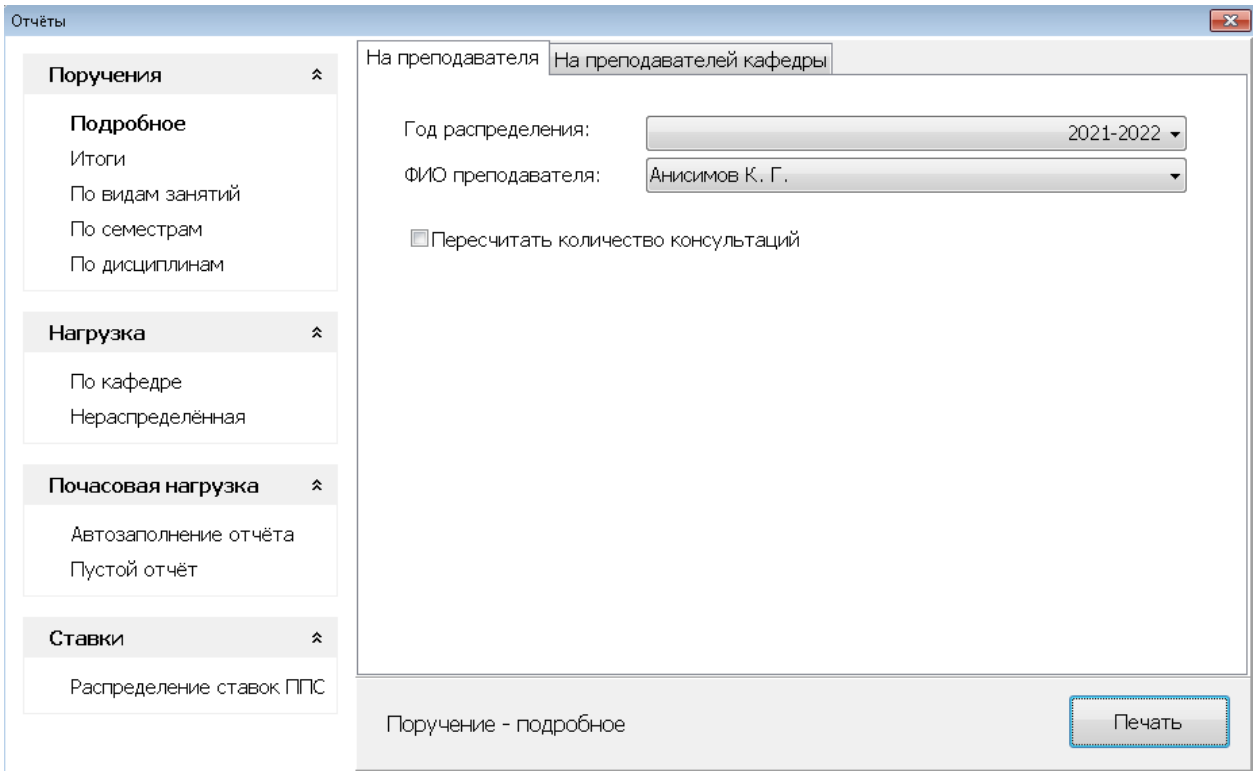
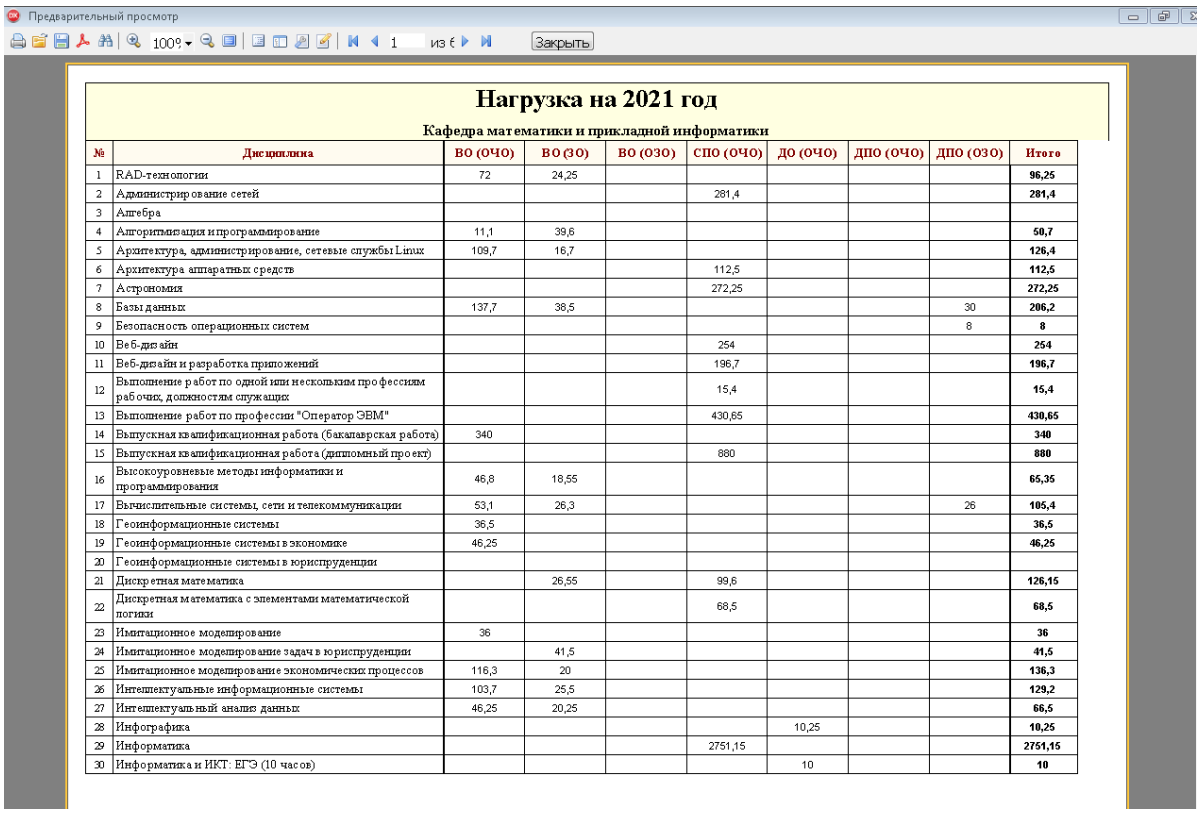


Рисунок 1.8 – Отчёты ИС «Поручения»



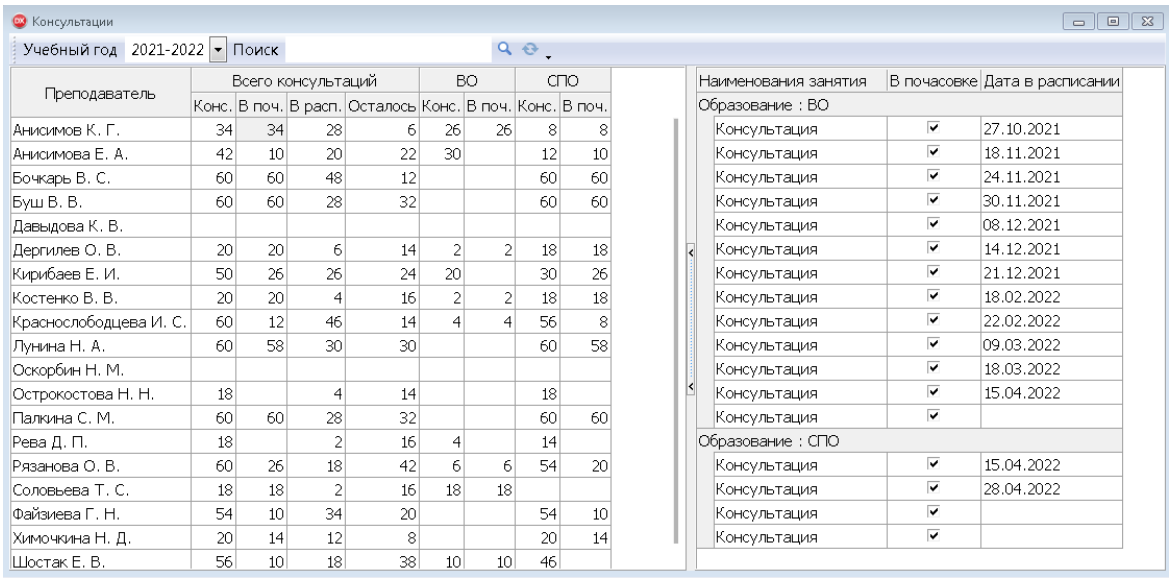
Нагрузка на 2021 год

Кафедра математики и прикладной информатики

№	Дисциплина	ВО (ОЧО)	ВО (ЗО)	ВО (ОЗО)	СПО (ОЧО)	ДО (ОЧО)	ДПО (ОЧО)	ДПО (ОЗО)	Итого
1	RAD-технологии	72	24,25						**96,25**
2	Администрирование сетей				281,4				**281,4**
3	Алгебра								
4	Алгоритмизация и программирование	11,1	39,6						**50,7**
5	Архитектура, администрирование, сетевые службы Linux	109,7	16,7						**126,4**
6	Архитектура аппаратных средств				112,5				**112,5**
7	Астрономия				272,25				**272,25**
8	Базы данных	137,7	38,5					30	**206,2**
9	Безопасность операционных систем							8	**8**
10	Веб-дизайн				254				**254**
11	Веб-дизайн и разработка приложений				196,7				**196,7**
12	Выполнение работ по одной или нескольким профессиям рабочих, должностям служащих				15,4				**15,4**
13	Выполнение работ по профессии "Оператор ЭВМ"				430,65				**430,65**
14	Выпускная квалификационная работа (бакалаврская работа)	340							**340**
15	Выпускная квалификационная работа (дипломный проект)				880				**880**
16	Высокоуровневые методы информатики и программирования	46,8	18,55						**65,35**
17	Вычислительные системы, сети и телекоммуникации	53,1	26,3					26	**105,4**
18	Геоинформационные системы	36,5							**36,5**
19	Геоинформационные системы в экономике	46,25							**46,25**
20	Геоинформационные системы в юриспруденции								
21	Дискретная математика		26,55		99,6				**126,15**
22	Дискретная математика с элементами математической логики				68,5				**68,5**
23	Имитационное моделирование	36							**36**
24	Имитационное моделирование задач в юриспруденции		41,5						**41,5**
25	Имитационное моделирование экономических процессов	116,3	20						**136,3**
26	Интеллектуальные информационные системы	103,7	25,5						**129,2**
27	Интеллектуальный анализ данных	46,25	20,25						**66,5**
28	Инфографика					10,25			**10,25**
29	Информатика				2751,15				**2751,15**
30	Информатика и ИКТ: ЕГЭ (10 часов)					10			**10**

Рисунок 1.9 – Отчет «Нагрузка по кафедре»

Кроме того, в информационной системе «Поручения» отображается информация о консультациях ППС кафедры: общее количество консультаций; количество часов в почасовой оплате; в расписании; оставшиеся часы консультаций, их распределение на ВО и СПО, а также дата в расписании (рисунок 1.10).



Учебный год 2021-2022 Поиск

Преподаватель	Всего консультаций				ВО		СПО	
	Конс.	В поч.	В расп.	Осталось	Конс.	В поч.	Конс.	В поч.
Анисимов К. Г.	34	34	28	6	26	26	8	8
Анисимова Е. А.	42	10	20	22	30		12	10
Бочкарь В. С.	60	60	48	12			60	60
Буш В. В.	60	60	28	32			60	60
Давыдова К. В.								
Дергилев О. В.	20	20	6	14	2	2	18	18
Кирибаев Е. И.	50	26	26	24	20		30	26
Костенко В. В.	20	20	4	16	2	2	18	18
Краснослободцева И. С.	60	12	46	14	4	4	56	8
Лунина Н. А.	60	58	30	30			60	58
Оскорбин Н. М.								
Острокостова Н. Н.	18		4	14			18	
Палкина С. М.	60	60	28	32			60	60
Рева Д. П.	18		2	16	4		14	
Рязанова О. В.	60	26	18	42	6	6	54	20
Соловьева Т. С.	18	18	2	16	18	18		
Файзиева Г. Н.	54	10	34	20			54	10
Химочкина Н. Д.	20	14	12	8			20	14
Шостак Е. В.	56	10	18	38	10	10	46	

Наименования занятия	В почасовке	Дата в расписании
Образование : ВО		
Консультация	☑	27.10.2021
Консультация	☑	18.11.2021
Консультация	☑	24.11.2021
Консультация	☑	30.11.2021
Консультация	☑	08.12.2021
Консультация	☑	14.12.2021
Консультация	☑	21.12.2021
Консультация	☑	18.02.2022
Консультация	☑	22.02.2022
Консультация	☑	09.03.2022
Консультация	☑	18.03.2022
Консультация	☑	15.04.2022
Консультация	☑	
Образование : СПО		
Консультация	☑	15.04.2022
Консультация	☑	28.04.2022
Консультация	☑	
Консультация	☑	

Рисунок 1.10 – Вкладка «Консультации» ИС «Поручения»

Из ИС «Поручения» и ИС «График учебного процесса» информация поступает в ИС «Редактор расписания занятий», где происходит понедельный расчет часов и создание расписание для студентов и преподавателей (рисунок 1.11).

Расписание отображается на образовательном портале Института и мобильном приложении «Расписание» для ОС Android (рисунок 1.12 – 1.13).

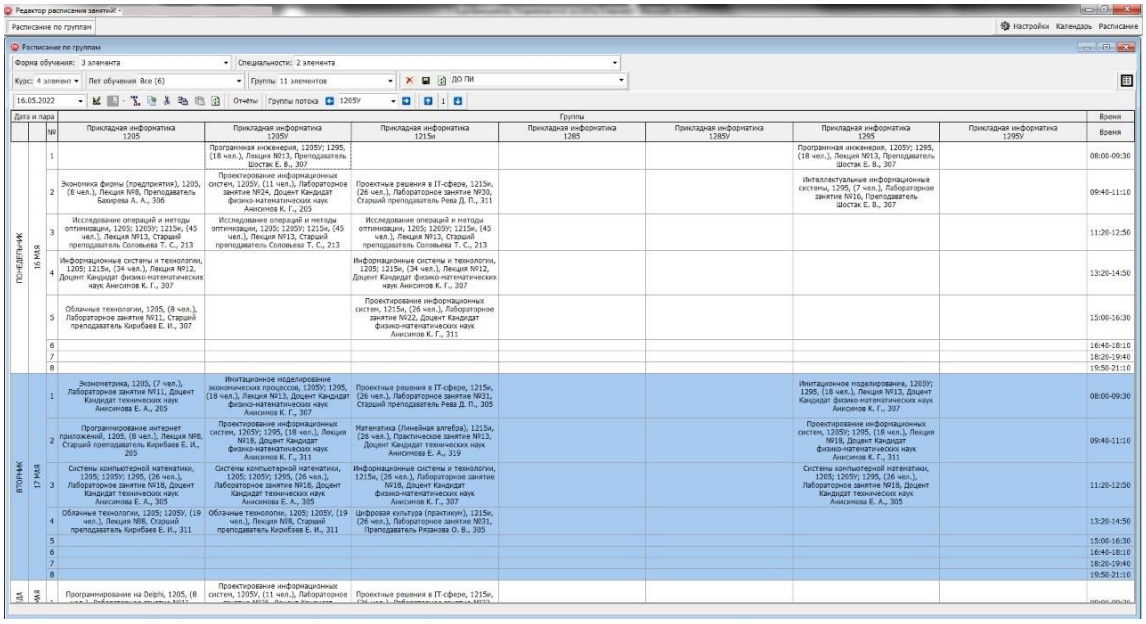


Рисунок 1.11 – Информационная система «Редактор расписания занятий»

Рисунок 1.12– Мобильное приложение «Расписание» для ОС Android

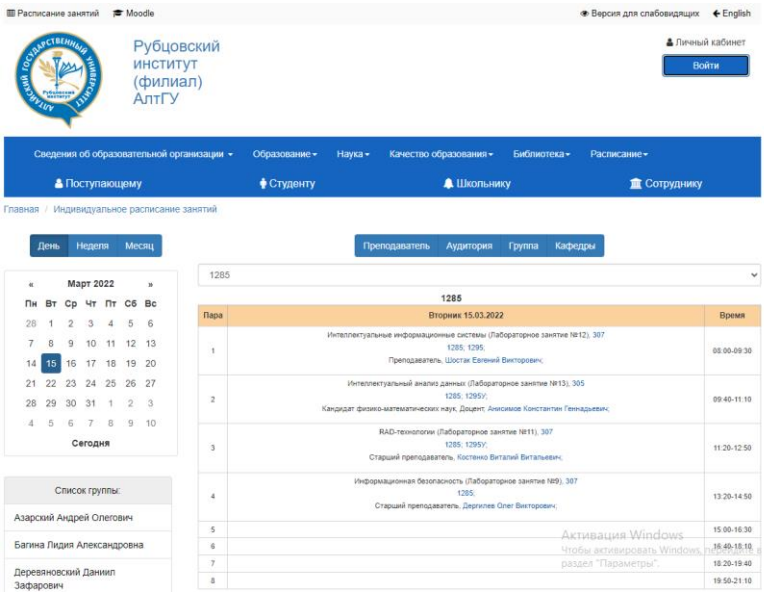


Рисунок 1.13 – Расписание занятий на образовательном портале Института

Информационная система «РПД/ФОС» предназначена для создания рабочих программ и фонда оценочных средств по дисциплинам учебных планов. Данная ИС связана с ИС «Учебные планы», ИС «Компетенции», ИС «Поручения».

Из ИС «Учебные планы» выбирается направление подготовки, профиль и название дисциплины, из ИС «Поручения» происходит закрепление преподавателей за дисциплинами». Из ИС «Компетенции» автоматически переносятся компетенции и индикатор их достижения (рисунок 1.14).

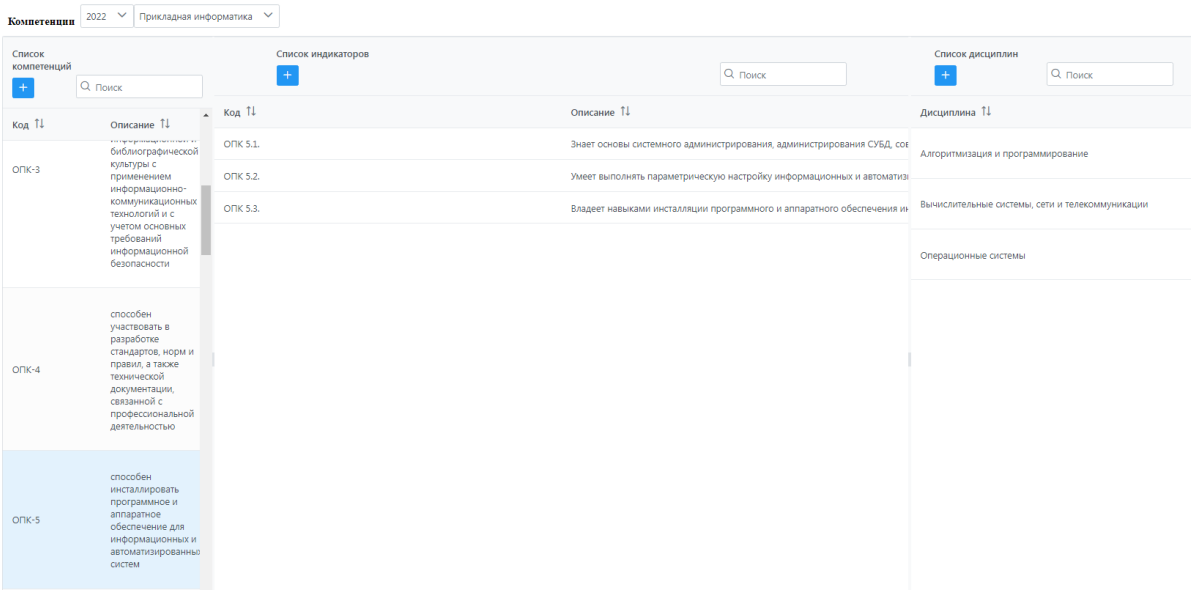


Рисунок 1.14 – Информационная систем «Компетенции»

При подаче абитуриентом заявления на поступление, технический секретарь приемной комиссии вносит данные поступающего в информационную систему «Абитуриент», такие, как «Основные сведения», «Контактная информация» и «Документы».

После процедуры зачисления данные из ИС «Абитуриент» копируются в ИС «Студенты», а затем из поступивших студентов формируются группы (рисунок 1.15).

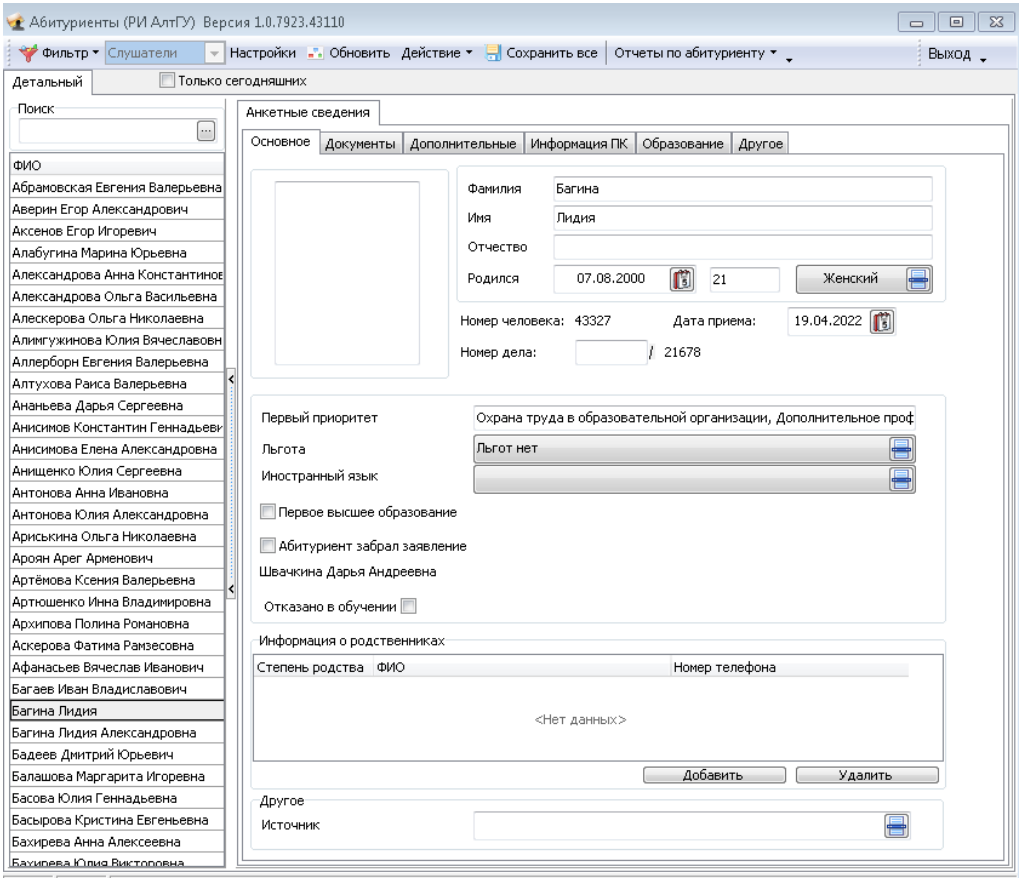


Рисунок 1.15 – Информационная система «Абитуриенты»

Информационная система «Студенты» представляет собой базу, в которой хранится информация о каждом обучающемся, выпускнике, отчисленном студенте, студенте в академическом отпуске, слушателе курсов ДПО, выпускнике курсов ДПО.

В ИС «Студенты» пользователь имеет возможность посмотреть анкетные данные студента, информацию об успеваемости, документообороте и т.д. (рисунок 1.16).

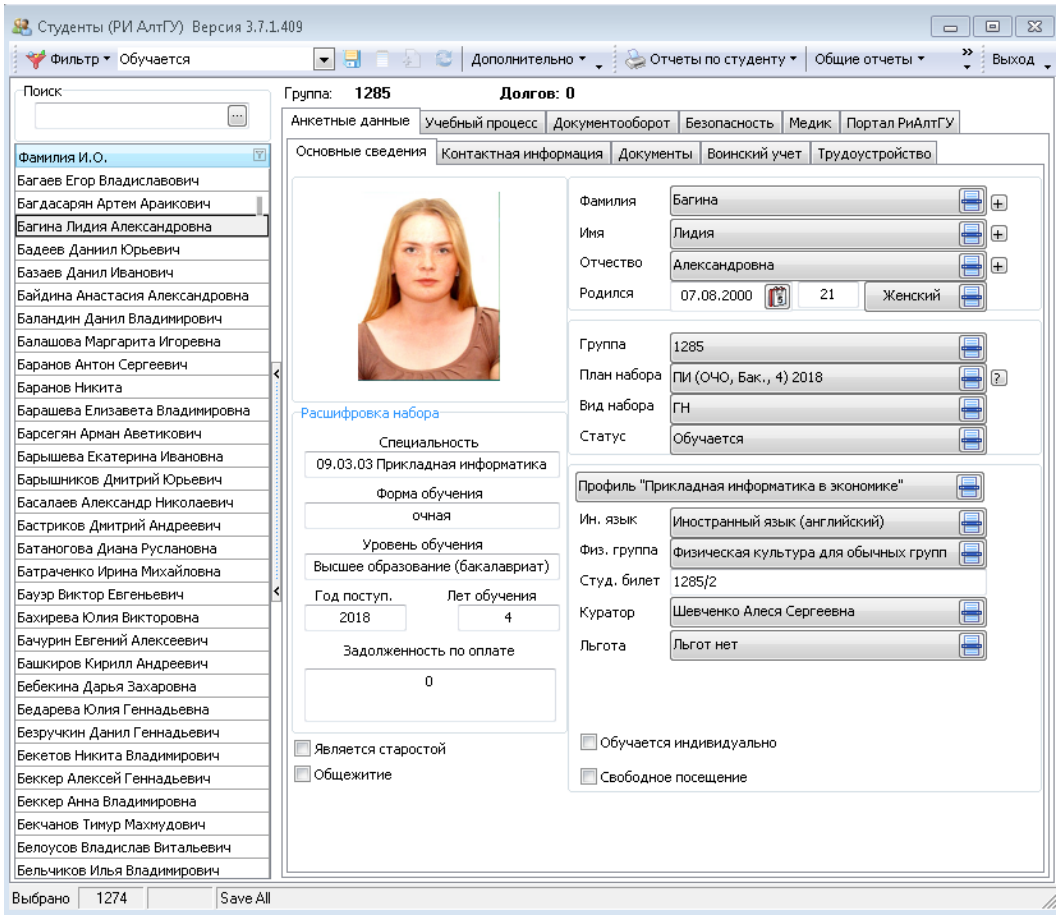


Рисунок 1.16 – Информационная система «Студенты»

Все существующие информационные системы, обеспечивающие учебный процесс функционируют следующим образом: на основании электронных рабочих учебных планов (ИС «Учебные планы»), графиков учебного процесса (ИС «Графики учебного процесса») и данных из ИС «Кадры» в ИС «Поручения» происходит закрепление часов за преподавателями кафедр; на основании закрепления преподавателей и расчета часов формируются планы-графики специальностей, которые выдаются на руки каждому студенту всех форм обучения в виде учебных поручений; на основании планов-графиков и графиков учебного процесса в ИС «Редактор расписания занятий» разрабатывается еженедельное расписание занятий всего Института.

На сегодняшний день, в Институте создан многоуровневый информационный комплекс для обеспечения учебного процесса. Данный

комплекс на основе информации, хранящейся в ИС «Кадры», «Абитуриенты», «Студенты», «Электронная библиотека», «1С: Предприятие 8.3.», позволяет автоматизировать все этапы управления учебным процессом, начиная от проектирования учебных планов и заканчивая формированием поручений ППС.

В настоящее время, в Институте существует следующая конфигурация серверов Института:

1. Файловый сервер RFAGU. Выполняет функции главного контроллера домена RBDOM, обеспечивающего централизацию управления ЛВС института, файлового сервера личных и общедоступных ресурсов, СПС и Консультант +. Технические характеристики: 2xP4Xeon 2,8GHz, 4x2Gb, IDE-SATA 500Gb, 1Tb, RAID5 SATA 3x1Tb, 2xGigabit Ethernet, установленное ПО – Gentoo Linux Server Edition.

2. Сервер WWW. Выполняет функции базового WEB-сервера со следующим набором сервисов: HTTP, FTP, DNS, MAIL, PROXY, APACHE, MySQL. Технические характеристики: 2xP4Xeon, 3,0GHz, 4x2Gb, RAID1 SATA 2x500Gb, 2xGigabit Ethernet. Установленное ПО – Gentoo Linux Server Edition.

3. Сервер GTW. Выполняет функции шлюза, обеспечивающего интеграцию сети института в глобальные сети Internet. Здесь установлен корпоративный межсетевой экран. Технические характеристики: Pentium-IV 2,4GHz, IDE-SATA 80Gb, 2xGigabit Ethernet. Установленное ПО – Gentoo Linux Server Edition.

4. Сервер EDUSERVER. Выполняет функции сервера БД для используемых в учебном процессе банковских информационных систем «RS-Bank 5.1». На нем установлен web-сервер Apache с модулем PHP 7.2.2, обеспечивающий онлайн доступ к расписанию занятий, система управления курсами Moodle, СКУД обеспечивающая автоматическое управления входом и выходом людей в здание организации, модуль FastReport, обеспечивает формирование отчетов печати для приложений. Технические

характеристики: Intel-I7 (4 core) 3,00GHz, 8Gb, IDE-SATA 500Gb, Gigabit Ethernet. Установленное ПО – MS Windows 2012 AS.

5. Сервер ORACLE. Выполняет функции сервера БД для хранения информации для портала и информации необходимой для учебного процесса. Технические характеристики: 2xP4Xeon, 2,8GHz, 24Gb, RAID5 SATA 4x200Gb, 2xGigabit Ethernet. Установленное ПО – Gentoo Linux Server Edition.

Установлено и используется следующее программное обеспечение:

- операционные системы (Windows 7 и Windows 10);
- офисные программы (Пакет Microsoft Office npo 2010);
- графические пакеты (Adobe Photoshop, corelDRAW);
- бухгалтерские системы (1С предприятие 8(8.3.10.2505);
- СУБД (Microsoft SQL Server 2016 Express LocalIDB, Microsoft SQL Server 2016 Transact-SQL Script Dom);
- средства программирования и разработки программного обеспечения (Java 8 Update 231 (64-bit), Cisco Packet Tracer 7.3.0 64Bit, Delphi 7);
- информационно-справочные системы (КонсультантПлюс ПРО).

Электронная информационно-образовательная среда и доступ к информационным системам реализуется посредством локальной вычислительной сети института.

Общий парк персональных компьютеров и мобильных систем доступа составляет около 300 ед., из них в учебном процессе используются более 200 ед., для осуществления образовательного процесса студентам предоставлено 11 компьютерных классов и абонентских систем, включая 4 класса на мобильных платформах.

Все административные подразделения, кафедры, компьютерные классы и лаборатории объединены в единую локальную вычислительную сеть, построенную на основе традиционных структурированных кабельных

системах (СКС), суммарной протяженностью около 12 км, со скоростью передачи данных в сетях подразделений 1Гбит/с.

Кроме этого, имеется 2 сегмента магистральных волоконно-оптических каналов связи, объединяющих между собой оба корпуса Рубцовского института (филиала) АлтГУ в пределах кампуса. Общая протяженность данной линии 1,58 км с пропускной способностью кабельной системы 1Гбит/с и возможностью расширения до 10 Гбит/с, за счет модернизации активного сетевого оборудования. Один оптический сегмент используется как опорный канал для доступа пользователей к ресурсам локальной сети института и внешним сетям, а другой для организации внутренней цифровой IP-телефонии на основе голосовых шлюзов VoIP.

1.2 Анализ функционирования объекта исследования

UML (англ. Unified Modeling Language – унифицированный язык моделирования) – язык графического описания для объектного моделирования в области разработки программного обеспечения, моделирования бизнес-процессов, системного проектирования и отображения организационных структур. UML является языком широкого профиля, это – открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью [7].

UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML-моделей возможна генерация кода[7].

Для функционального и объектного описания корпоративной информационной сети Института был выбран язык моделирования UML.

Для моделирования предметных областей используются следующие

виды диаграмм:

1. Диаграмма классов – предлагает визуальное представление о классах и о том, каким образом они связаны.

2. Диаграмма объектов – показывает взаимосвязи системных объектов и о потенциальных недостатках, которые необходимо исправить, предлагает наилучшее представление.

3. Диаграмма компонентов – показывает взаимосвязи логических групп элементов.

4. Составная структурная диаграмма – показывает взаимодействие между несколькими классами внутренней структуры.

5. Диаграмма развертывания – показывает программные и аппаратные компоненты и их взаимосвязи.

6. Диаграмма пакетов – показывает взаимосвязи крупных компонентов, образующих сложную систему.

7. Диаграмма профиля – путем определения пользовательских стереотипов, теговых значений, ограничений помогает создавать семантику и новые свойства для диаграмм UML.

8. Диаграмма деятельности – изображает набор выполнения операций для достижения цели.

9. Диаграмма вариантов использования – набор происходящих событий, когда кто-либо или что-либо, взаимодействующее с системой из-за пределов системы, использует ее для завершения процесса.

10. Обзорная диаграмма взаимодействия – диаграмма деятельности, составленная из разных диаграмм взаимодействия.

11. Временная диаграмма – функционально показывает действие объектов и операторов на временной шкале.

12. Диаграмма конечного автомата – помогает описать поведение объекта и его изменение в зависимости от внутренних и внешних событий.

13. Диаграмма последовательности – в хронологическом порядке показывает последовательность сообщений и взаимодействий между

операторами и объектами.

14. Диаграмма связи – показывает организацию объектов, участвующих во взаимодействии.

Для описания структуры всей информационной системы в целом, была использована диаграмма компонентов (см. Приложение А).

Диаграмма компонентов (англ. Component diagram) – элемент языка моделирования UML, статическая структурная диаграмма, которая показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами.

В качестве физических компонентов могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты и т.п.

Для построения диаграммы компонентов использовалось CASE средство UML Designer – графический инструмент для редактирования и визуализации моделей UML. Данное программное обеспечение является бесплатным продуктом компании Obeo.

В ходе анализа и моделирования объекта исследования были выявлены следующие проблемы:

– отсутствие единой точки идентификации и авторизации;

– отсутствие механизма наделения пользователя клеймсами и скоупами, для работы с КИС Института;

– отсутствие механизма интеграции приложений в личный кабинет сотрудника.

1.3 Определение цели и задач проектирования web-приложения

Разрабатываемое web-приложение интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ на основе подхода micro-frontend предназначено для централизации

внутренней корпоративной информации и организации единой точки доступа к корпоративным ресурсам Института.

Процесс web-интеграции заключается в создании корпоративного информационного портала путем сведения воедино компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ.

Портал представляет собой окно к структурированным, персонифицированным, корпоративным и другим данным, доступ к которым осуществляется посредством Web-интерфейса.

Внедрение web-приложения позволит достичь следующих положительных сторон:

1. Единая точка идентификации и авторизации.
2. Представление информации при помощи web-браузера.
3. Возможность организации удаленной работы с корпоративной информационной системой Института.
4. Беспрерывная поставка обновлений ИС, а также удобное внедрение новых информационных систем в КИС.
5. Единая система оповещения и информирования сотрудников.

Перечисленные положительные факторы от внедрения web-приложения дают возможность поставить главную цель данного исследования:

– разработать web-приложение интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ.

1.4 Обзор и анализ существующих разработок

На рынке программных продуктов отсутствуют решения, которые позволили бы интегрировать компоненты корпоративной информационной

системы Рубцовского института (филиала) АлтГУ. Это связано с тем, что компоненты КИС являются собственной разработкой Института.

1.5 Выбор и обоснование проектных решений

1.5.1 Выбор фреймворков, технологий разработки и технологических платформ

Основным инструментом для создания пользовательского интерфейса web-приложения стал JavaScript фреймворк Vue.

Работа над фреймворком Vue.js началась в 2013 году сотрудником Google Эваном Ю (Evan You). Релиз Vue состоялся в феврале 2014 года. Версия 1.0 была выпущена в октябре 2015 года, а версия 2.0 – в сентябре 2016.

Основные преимущества фреймворка:

1. Vue во время отрисовки автоматически отслеживает зависимости компонентов, поэтому фреймворк способен самостоятельно определить, какие компоненты действительно необходимо повторно отрисовать.

2. Все дополнительные библиотеки Vue (библиотеки управления состоянием приложения Vuex и библиотеки роутинга Vue Router) имеют официальную поддержку от разработчиков фреймворка.

3. Vue имеет инструменты для генерации проектов с использованием командной строки (Vue CLI). Это упрощает создание новых проектов с использованием подходящей системы сборки (Webpack, Browserify или без таковой).

В качестве библиотеки управления состоянием для Vue.js выбрана Pinia. Pinia имеет уникальные возможности для управления хранимыми данными, такие как: расширяемость, организация модулей хранилищ, группировка изменений состояния, создание нескольких хранилищ и т.д.

Для получения и отображения данных из API используется клиентская библиотека axios, основанная на Promise HTTP-клиента.

Web-приложение выполнено на основе архитектуры микрофронтенд (micro-frontend), которая была предложена ThoughtWorks в 2016 г.

Концепция микро-интерфейса опирается на архитектурную концепцию микросервисов (micro-zoe) [8].

Микрофронтенд (micro-frontend) — архитектурный подход, в котором независимые приложения собраны в одно большое приложение. Он дает возможность объединить в одном приложении разные виджеты или страницы, написанные разными командами с использованием разных фреймворков (рисунок 1.17).

Главные преимущества микрофронтенд-подхода:

1. Модульность – возможность в облегчении понимания, разработки и тестировании, а также обеспечение устойчивости к эрозии архитектуры.
2. Нулевая зависимость – отсутствие зависимостей, что дает небольшой размер и высокую масштабируемость.
3. Масштабируемость – поскольку микросервисы реализуются и развертываются независимо друг от друга, то есть выполняются в рамках независимых процессов, их можно контролировать и масштабировать независимо.
4. Совместимость со всеми фреймворками – возможность нормальной работы в любой технической среде.

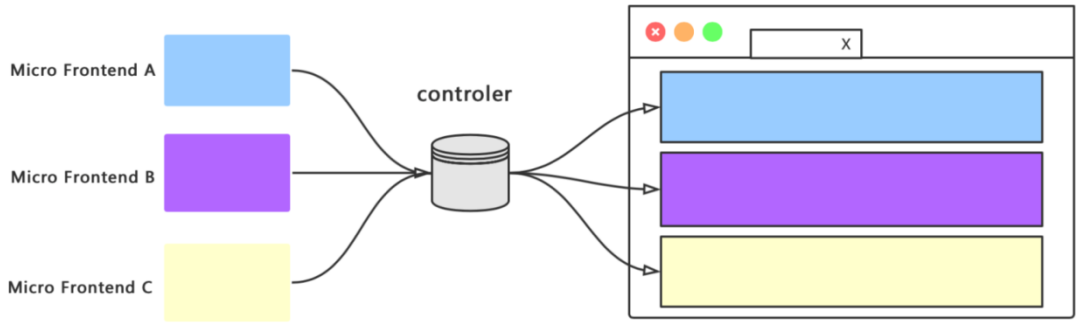


Рисунок 1.17 – Архитектура micro-frontend

1.5.2 Выбор средств разработки и проектирования

В качестве среды разработки web-приложения выступило решение от компании JetBrains – WebStorm.

WebStorm представляет собой удобный и умный редактор для JavaScript, HTML и CSS, с подсветкой и автодополнением кода, проверкой его на наличие ошибок, а также поддерживающий фреймворки Angular, React и Vue.js.

Основные возможности:

1. Умный редактор:

– автодополнение кода – формирование подсказок с учетом контекста и типов;

– анализ качества кода – выявление синтаксических и лексических ошибок и опечаток в коде;

– безопасные рефакторинги – переименование файлов, папок и символов, извлечение компонентов, методов и переменных;

– быстрый просмотр документации для символа.

2. Встроенные инструменты для разработчиков, такие как: отладка JavaScript, юнит-тестирование, продвинутая интеграция с VCS, локальная история изменений (отслеживание всех изменений в рамках проекта и их отмена, даже если проект не подключен к системе контроля версий), встроенный HTTP-клиент, поддержка линтеров, встроенный терминал.

3. Навигация и поиск:

– удобная навигация – использование окна Search Everywhere для поиска действий, файлов, классов и символов;

– исследование кода – возможность быстрого перехода к объявлению символа;

– навигация по проекту – просмотр файлов проекта и переход к недавно открытым или редактированным файлам и фрагментам кода;

– поиск текста – нахождение и замена текстовых строк в определенных файлах и во всем проекте.

4. Совместная работа – эффективная совместная удаленная разработка и парное программирование.

5. Кастомизация. Возможность индивидуальной настройки рабочей среды – тем интерфейса, расположения элементов, сочетания горячих клавиш.

В WebStorm имеется лицензионный ключ для бесплатного пробного периода использования – 30 дней.

Для данного проекта выбрана бесплатная академическая лицензия для студентов, без ограничений функционала.

2 Проектная часть

2.1 Разработка функционального обеспечения

Построенная компонентная диаграмма «как есть» в подразделе «Анализ функционирования объекта исследования» аналитической части, и выявленные недостатки приводят к необходимости построения модели «как должно быть». Задачей описания модели «как должно быть», является нахождение мер блокирования отрицательного влияния неудовлетворительных бизнес-факторов, найденных при анализе. Полученная компонентная диаграмма указана в Приложении Б.

Модель «как должно быть» полностью устраняет недостатки действующей модели:

– отсутствие единой точки идентификации и авторизации;

– отсутствие механизма наделения пользователя клеймсами и скоупами, для работы с КИС Института;

– отсутствие механизма интеграции приложений в личный кабинет сотрудника.

2.2 Разработка информационного обеспечения

2.2.1 Используемые классификаторы и системы кодирования

Порт представляет собой числовой идентификатор программы или процесса, который обслуживает сетевые соединения на заданном IP-адресе.

Таким образом, при внедрении нового подприложения в родительское приложение, ему автоматически присваивается определенный порядковый номер порта.

2.2.2 Характеристика нормативно-справочной и входной оперативной информации

В качестве входной информации в web-приложении будет выступать информация об интегрируемом подприложении (рисунок 2.1):

– наименование;

– доменное имя и номер порта;

– наименования callback функций для привязки хуков жизненного цикла подприложения.

```
<template>
  <div>
    <ProgressSpinner v-show="loading"/>
    <micro-app name="workingprogram"
               keep-alive
               url="https://workprogramm.rb.asu.ru:4011/"
               baseroute="/workingprogram"
               :data='microAppData'
               @created='handleCreate'
               @beforemount='handleBeforeMount'
               @mounted='handleMount'
               @unmount='handleUnmount'
               @error='handleError'
               @datachange='handleDataChange'
               disableScopecss
               v-show="!loading"
    ></micro-app>
  </div>
</template>
```

Рисунок 2.1 – Пример интеграции подприложения

2.2.3 Характеристика результатной информации

В результате построения диаграмм вариантов использования (use case diagram) «как есть» (рисунок 2.2) и «как должно быть» (рисунок

2.3), наглядно видно разницу в процедуре входа в систему сотрудника – пользователя корпоративной информационной системы Института.

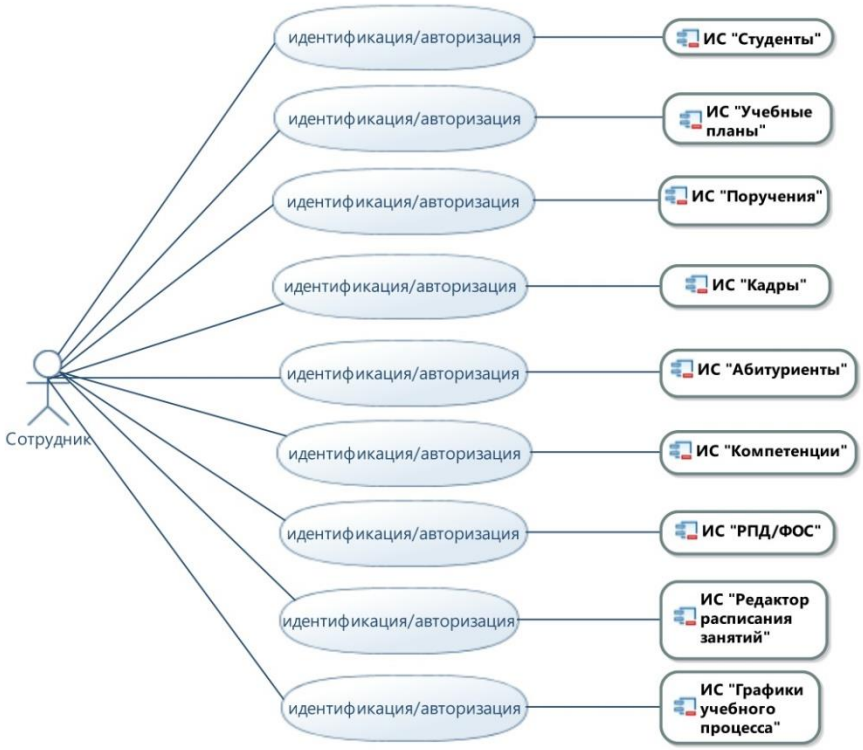


Рисунок 2.2 – Диаграмма вариантов использования «как есть»

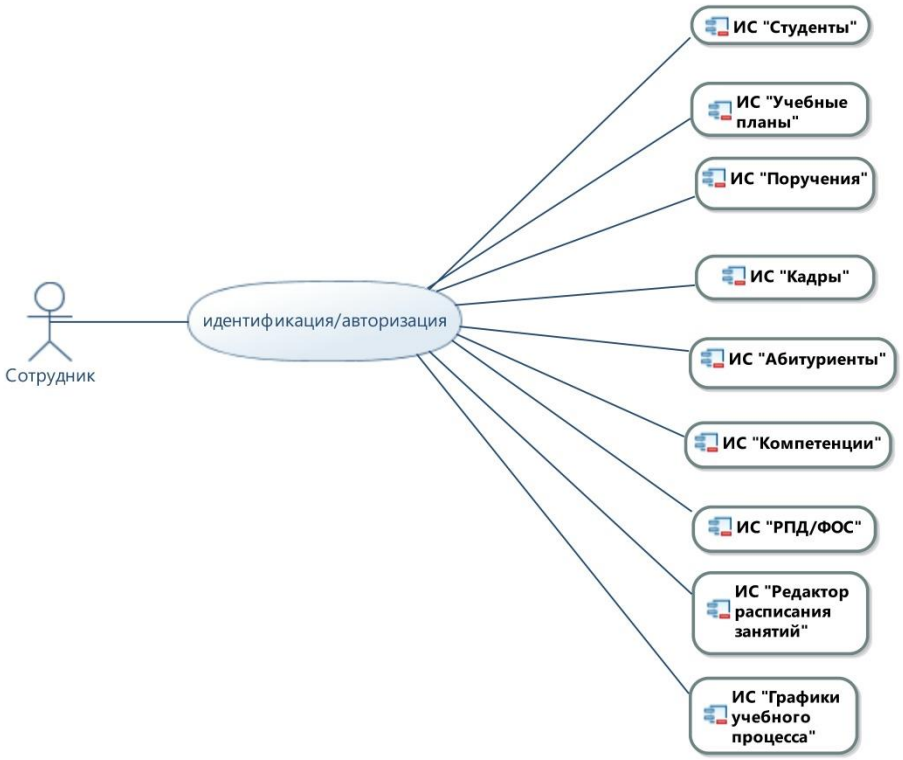


Рисунок 2.3 – Диаграмма вариантов использования «как должно быть»

При существующей системе доступа, пользователю необходимо при входе в информационные системы каждый раз вводить свой пароль доступа. При новой же системе, пройдя единожды процедуру авторизации и идентификации, пользователь может в течение сеанса работы обращаться к компонентам КИС, при этом повторная авторизации и идентификации не требуется.

2.3 Разработка программного обеспечения

2.3.1 Описание клиентских программных модулей

Для реализации microfrontend-а был использован пакет micro-zoe/micro-app. С использованием данного пакета, были написаны две основные части будущих приложений:

1. Родительское приложение.
2. Шаблон для разработки дочерних приложений.

Главная задача родительского приложения заключается в том, что оно должно агрегировать все дочерние, которые указаны у него в шаблонах (рисунок 2.1). Так же при запуске оно должно пройти авторизацию клиента (самого приложения) и после этого предоставить возможность идентификации/авторизации пользователя, последующего получения скоупов и клеймсов для проксирования их в дочерние приложения, а так же рендера главного меню (с ограничениями для пользователя полученными после его авторизации).

На рисунке 2.4 представлен программный код инициализации запуска родительского приложения. На 8 строке рисунка 2.4 происходит подключение пакета micro-zoe/micro-app, после этого на 28 строке выполняется запуск прослушивателя (компонент рендера их в HTML5 Component) дочерних приложений. На строках с 9-19 приложение при

помощи сервисов accountService и clietnAuthorizationService родительское приложение авторизует себя и пользователя на oAuth сервере. На 21 сроке того же рисунка создаем callback функцию, которая вызовет создание экземпляра Vue с последуеющем его запуском в среде mainOidc, которая контролирует распределение скоупов и клеймсов пользователя при работе с приложением.

```
import { createApp } from 'vue'
import App from './App.vue'
import prime from './config/primevue'
import './assets/sass/layout.sass'
import router from './router'
import { createPinia } from 'pinia'
import mainOidc from '@/service/oAuth'
import microApp from '@micro-zoe/micro-app'
import { accountService } from '@/service/accountAuthorization.service'
import { clientAuthorizationService } from '@/service/clientAuthorization.service'

try {
  (async () => {
    const resultAuthClient = await clientAuthorizationService.login()
    const resultAuthUser = await accountService.refresh()
  })()
} catch (e) {
  router.push({ path: '/login' })
}
microApp.start()
mainOidc.startup().then(ok => {
  if (ok) {
    const app = createApp(App).use(router).use(prime).use(createPinia())
    app.config.globalProperties.$oidc = mainOidc
    app.mount( rootContainer: '#app')
  }
}
)
```

Рисунок 2.4 – Программный код инициализации запуска родительского приложения

На рисунке 2.5 представлен программный код инициализации запуска дочернего приложения. Первичная задача состоит в том, что необходимо

подключить пакеты, которые необходимы для запуска (2 строка), роутинга (5 строка, стейтменеджмента (6 строка) приложения.

```
1  import './public-path'
2  import { createApp, App as AppInstance } from 'vue'
3  import { createRouter, createWebHistory, Router, RouterHistory, RouterOptions } from 'vue-router'
4  import App from './App.vue'
5  import routes from './router/index'
6  import { createPinia } from 'pinia'
7  import primevue from '@/config/prime'
```

Рисунок 2.5 – Программный код инициализации запуска дочернего приложения

После этого необходимо задекларировать хранение переменных, для дочернего приложения (рисунок 2.6). Это необходимо для хранения данных передаваемых родительскому приложению при вызове callback функций при срабатывании хуков

```
9  declare global {
10   interface Window {
11     microApp: any
12     __MICRO_APP_NAME__: string
13     __MICRO_APP_ENVIRONMENT__: string
14     __MICRO_APP_BASE_ROUTE__: string
15   }
16 }
```

Рисунок 2.6 – Регистрация глобальных переменных для дочернего приложения

После этого дочернее приложение должно начать отслеживать состояние среды в которую его пытаются подгрузить, если оно совпадает с заложенной микроинтерфейсной средой в пакете micro-app, то оно начинает дожидаться данных от родительского приложения и в дальнейшем после получения базового пути начнет перенаправлять (подменять url в адресной строке браузера) себя уже на скорректированный адрес с использованием базового пути (рисунок 2.7) . Ввиду ускоренной разработки пакета micro-app,

были непредусмотренны некоторые аспекты базового пакета роутинга vue-route4, поэтому необходимо исправить их совместимость по средству корректировки перезаписи адреса в адресной строке и истории браузера, что бы пользователи могли использовать кнопку назад (рисунок 2.8).

```
// Взаимодействие с родительским приложением
function handleMicroData (router: Router) {
  // Является ли это микроинтерфейсной средой
  if (window.__MICRO_APP_ENVIRONMENT__) {
    // Активно получаем данные, отправленные родительским приложением
    console.log('child-workingprogram getData:', window.microApp.getData())
    // Мониторинг изменений данных, отправляемых родительским приложением
    window.microApp.addDataListener((data: Record<string, unknown>) => {
      console.log('child-workingprogram addDataListener:', data)
      // Перейти, когда родительское приложение отправит путь
      if (data.path && data.path !== router.currentRoute.value.path) {
        router.push(data.path as string)
      }
    })
    // отправляем данные в родительское приложение
    setTimeout( handler: () => {
      window.microApp.dispatch({ myname: 'child-workingprogram' })
    }, timeout: 3000)
  }
}
```

Рисунок 2.7 – Получение базового пути от родительского приложения

```
/**
 *  Он используется для решения проблемы, заключающейся в том, что когда основное приложение и подприложение
 * являются vue-router4, они конфликтуют друг с другом, что приводит к проблеме нажатия кнопки
 * возврата в браузере и ошибке маршрутизации..
 */
function fixBugForVueRouter4 (router: Router) {
  if (window.location.href.includes('/asu') || window.location.href.includes('/main-vite')) {
    /**
     * Важно:
     *  1、Основной маршрут, выдаваемый основным приложением: `/main-xxx/app-vue3`, из которых `/main-xxx`
     * является основным маршрутом основного приложения, которое необходимо удалить. Берем только
     * `/childPath`, разные проекты настраиваются в соответствии с реальной ситуацией
     */
    const realBaseRoute = window.__MICRO_APP_BASE_ROUTE__.replace( searchValue: /^\/asu[^/]+/g, replaceValue: '')
    router.beforeEach( guard: () => {
      if (typeof window.history.state?.current === 'string') {
        window.history.state.current = window.history.state.current.replace(new RegExp(realBaseRoute, flags: 'g'), '')
      }
    })
    router.afterEach( guard: () => {
      if (typeof window.history.state === 'object') {
        window.history.state.current = realBaseRoute + (window.history.state.current || '')
      }
    })
  }
}
```

Рисунок 2.8 – Устранение ошибки приложения при работе с пакетом vue-route4 при обработки базового url родительского приложения

На рисунках 2.9 – 2.10 уже происходит подмонтирование и запуск дочернего приложения в микроинтерфейсной среде родительского приложения. Так же происходит привязка хуков mount и unmounts, при помощи которых можно отслеживать изменение состояния дочернего приложения в родительском приложении.

```
let app: AppInstance | null = null
let router: Router | null = null
let history: RouterHistory | null = null
function mount () {
  history = createWebHistory( base: window.__MICRO_APP_BASE_ROUTE__ || process.env.BASE_URL)
  router = createRouter( options: {
    history,
    routes
  }) as Router

  // @ts-ignore
  app = createApp(App)
  app.use(router)
  app.use(primevue)
  app.use(createPinia())
  app.mount( rootContainer: '#childWorkingProgram')

  console.log('Микро приложение child-vue3 предоставленный')

  handleMicroData(router)

  fixBugForVueRouter4(router)
}
```

Рисунок 2.9 – Создание экземпляра vue

```
// Поместите операцию размонтирования в хук onmount
function unmount () {
  app?.unmount()
  history?.destroy()
  app = null
  router = null
  history = null
  console.log('Микро приложение child-workingprogramm деинсталлирован')
}

// зарегистрируем хуки mount и onmount
if (window.__MICRO_APP_ENVIRONMENT__) {
  // @ts-ignore
  window[`micro-app-${window.__MICRO_APP_NAME__}`] = { mount, unmount }
} else {
  // отрендерим если надо запустить без microfrontend-а
  mount()
}
```

Рисунок 2.10 – Регистрация хуков mount unmount

2.3.2 Компоненты пользовательского интерфейса

Пользовательский интерфейс представляет собой коммуникационный канал, реализующий взаимодействие пользователя с программным обеспечением с помощью элементов управления.

При первом запуске веб-приложения перед пользователем отображается окно авторизации для входа в систему, которое содержит область ввода конфиденциальной информации (логин и пароль) (рисунок 2.11).

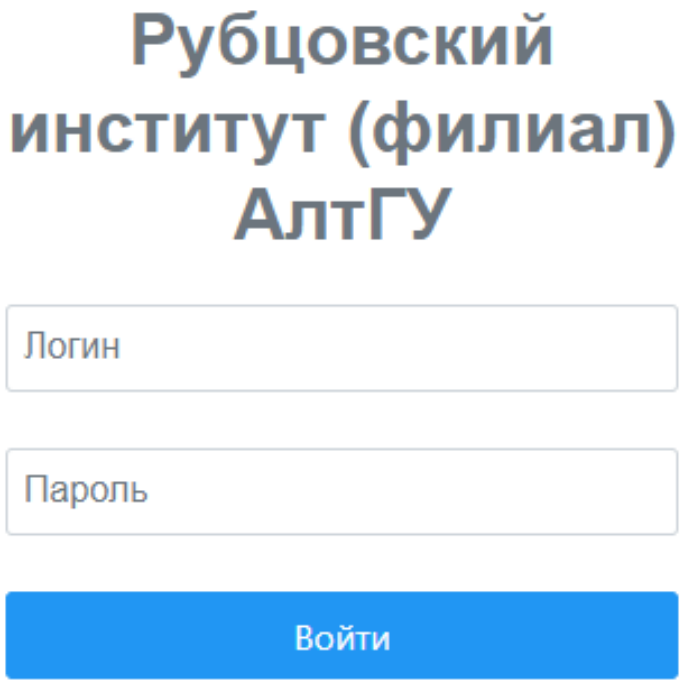


Рисунок 2.11 – Авторизация пользователя

После авторизации, пользователь попадает на страницу главного меню, внешний вид которого изображен на рисунке 2.12. Слева в меню располагаются интегрированные на данный момент ИС «Компетенции» (рисунок 2.13) и ИС «РПД/ФОС» (рисунок 2.14).

Рисунок 2.12 – Главное меню

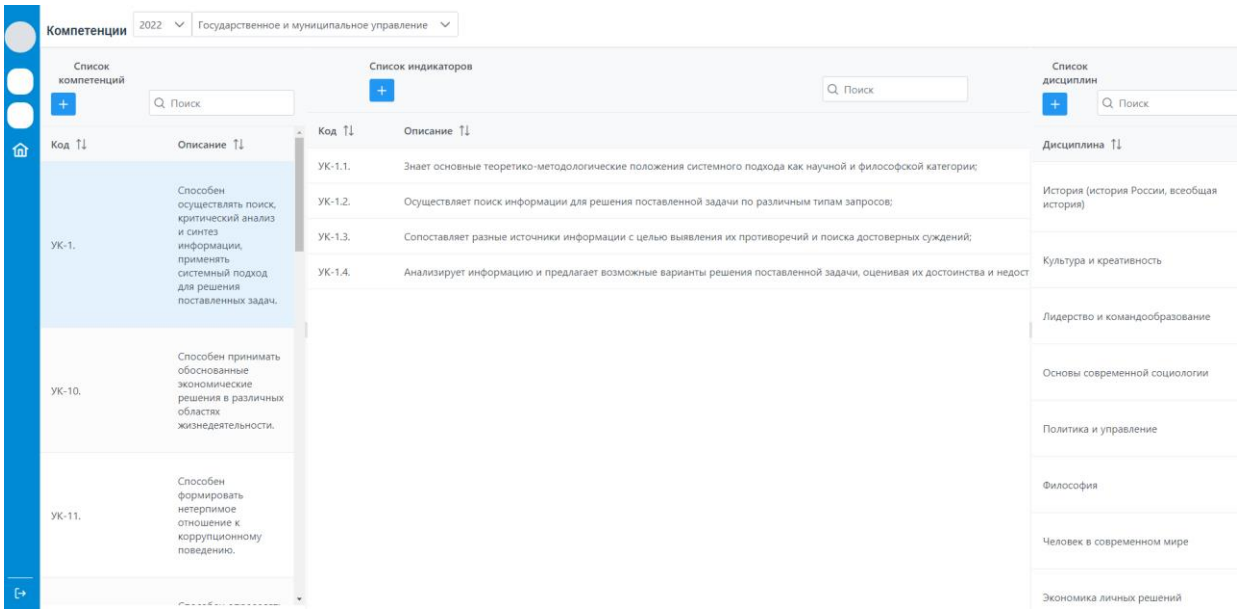


Рисунок 2.13 – ИС «Компетенции»

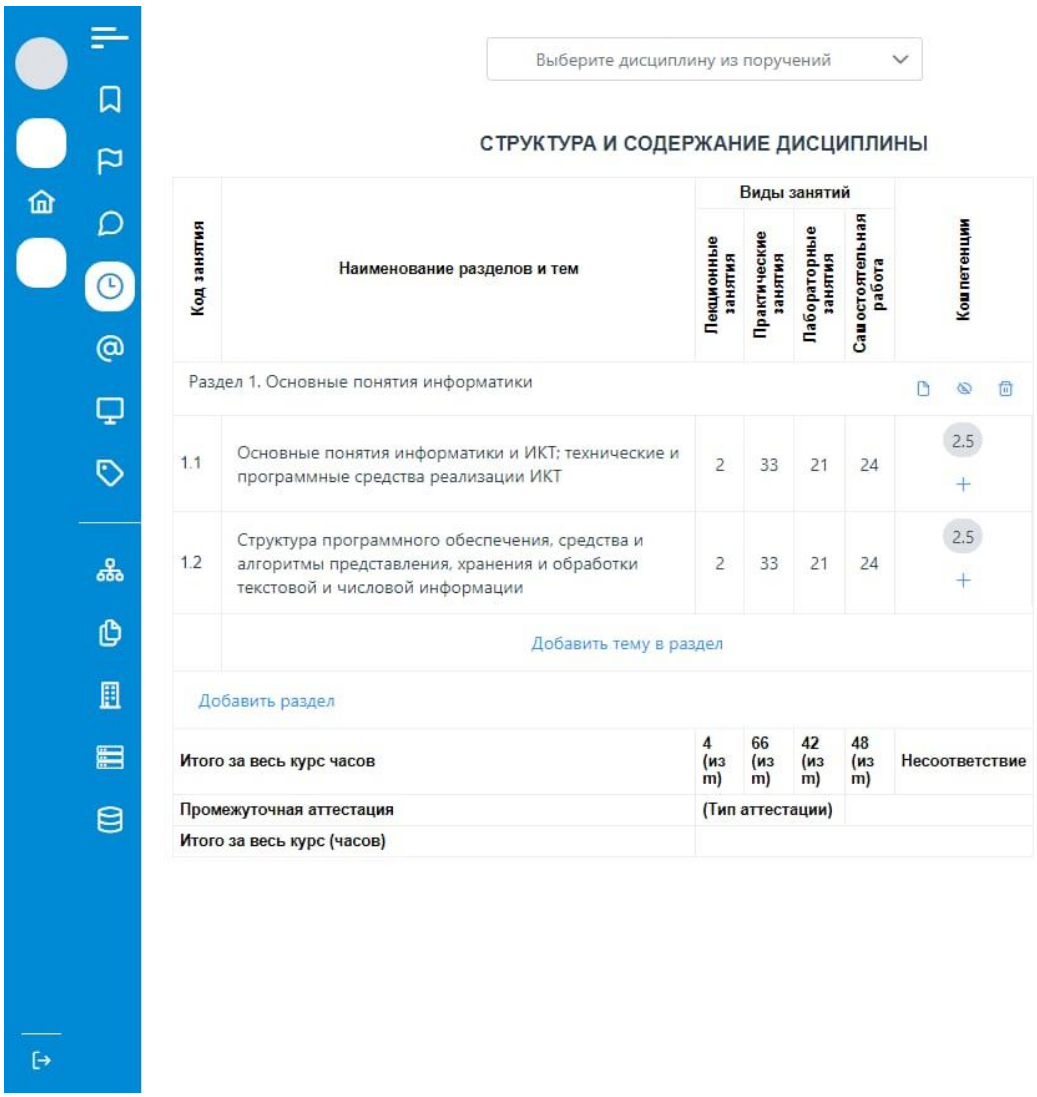


Рисунок 2.14 – ИС «РПД/ФОС»

Как только пользователь нажмет на кнопку выхода из системы, будет осуществлен переход на страницу авторизации.

2.4 Компьютерно-сетевое обеспечение

Серверная часть web-приложения использует существующее сетевое оборудование Рубцовского института (филиала) АлтГУ:

1. Сервер WWW. Выполняет функции базового WEB-сервера со следующим набором сервисов: HTTP, FTP, DNS, MAIL, PROXY, APACHE, MySQL. Технические характеристики: 2xP4Xeon, 3,0GHz, 4x2Gb, RAID1

SATA 2x500Gb, 2xGigabit Ethernet. Установленное ПО – Gentoo Linux Server Edition.

2. Сервер ORACLE. Выполняет функции сервера БД для хранения информации для портала и информации необходимой для учебного процесса. Технические характеристики: 2xP4Xeon, 2,8GHz, 24Gb, RAID5 SATA 4x200Gb, 2xGigabit Ethernet. Установленное ПО – Gentoo Linux Server Edition.

2.5 Обеспечение информационной безопасности

Под информационной безопасностью подразумевается состояние защищённости информации от преднамеренного или случайного несанкционированного доступа, хищения, модификации и уничтожения информации.

Для защиты от постороннего вторжения предусматриваются определенные меры безопасности. В ИС это осуществляется программными средствами, которые выполняют следующие функции:

- идентификация субъектов и объектов;
- разграничение доступа к ресурсам и информации;
- контроль и регистрация действий.

Процедура идентификации и подтверждения подлинности предполагает проверку является ли субъект, осуществляющий доступ, или объект, к которому осуществляется доступ, тем за кого себя выдает. Здесь используются следующие методы:

- простые, сложные или одноразовые пароли;
- средства анализа индивидуальных характеристик субъекта (геометрических данных);
- ключи, жетоны, магнитные карты;
- обмен вопросами и ответами с администратором системы;

– специальные идентификаторы и контрольные суммы для программ и данных.

После процедуры идентификации пользователь получает доступ к системе, где защита от несанкционированного доступа реализуется на трех уровнях:

– на уровне аппаратуры;
– на уровне программного обеспечения;
– на уровне данных.

Защита информации на 1 и 2 уровнях предусматривает управление доступом к различным вычислительным ресурсам (отдельным устройствам, операционной системы, служебным или личным программам пользователя).

Защита информации на уровне данных направлена на защиту информации в процессе обращения к ней, в процессе работы с файловой структурой ЭВМ, в процессе передачи информации по каналам связи [5].

В общем, комплекс программно-технических средств и организованных решений по защите информации от несанкционированного доступа характеризуется следующими действиями:

– логическое управление доступом;
– регистрацией, контролем и учетом работы;
– применением криптографических средств;
– обеспечением целостности информации.

Выделяют следующие формы контроля и управления доступом:

– предотвращение доступа к жесткому диску, каталогам и файлам;
– установка привилегий к группам файлов;
– защита от модификаций и изменений;
– защита от уничтожения;
– предотвращение копирования.

Под средствами защиты от копирования понимается такие средства, которые обеспечивают выполнение программой своих функций только при

опознании некоторого уникального не копированного элемента. Таким элементом может быть ключевой сменный носитель, часть оборудования или специальное устройство, подключаемое к компьютеру [6].

Защита от копирования реализуется выполнением ряда функций:

– идентификация среды, из которой запускается система;
– проверка подлинности среды, из которой произошел запуск;
– немедленная реакция на запуск из несанкционированной среды;
– обязательная регистрация санкционированного копирования;
– противодействие изучению алгоритмов работы системы.

Схема получения разрешения приложения для доступа к файлам сервера представлена на рисунке 2.15.

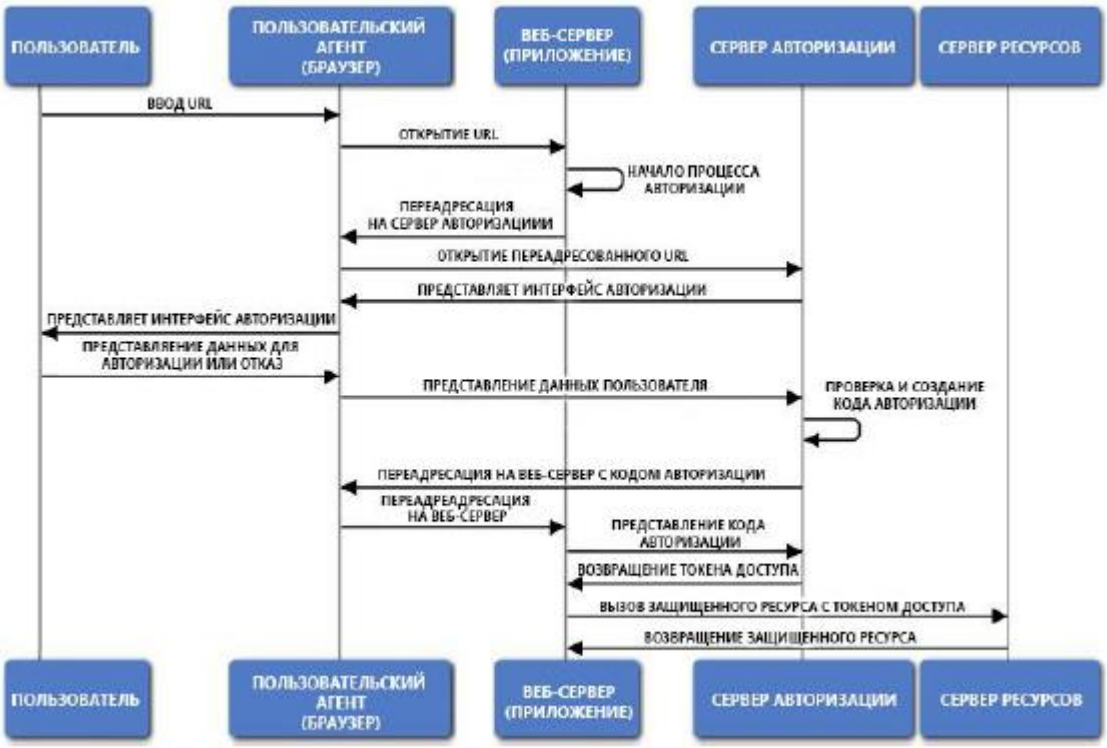


Рисунок 2.15 – Процесс авторизации по протоколу OAUTH 2.0

Главной особенностью схемы авторизации Oauth 2.0 является то, что клиентское приложение не имеет прямого доступа к авторизационным и идентификационным данным пользователя, а сам процесс авторизации

происходит на сервере авторизации Рубцовского института (филиала) АлтГУ [4].

Сам процесс обмена данными и идентификации пользователя в клиентском приложении, происходит при помощи токенов и ключей приложения ACCEESSTOKEN, RESOURCETOKEN и REFRESHTOKEN, а так же ключей PUBLICKEY и SECRETKEY выданных при регистрации приложения.

Процесс авторизации пользователя и получения данных имеет несколько этапов, а именно:

1. Инициализации процесса авторизации с посыла публичного ключа приложения к OAUTH-SERVER по определенному URL.
2. Пользователь загружается стандартная форма авторизации на ресурсах поставщика данных.
3. После ввода авторотационных и идентификационных данных пользователя, OAUTH-SERVER производит его авторизацию, и в случае успешного исхода авторизует пользователя и создает ACCESSTOKEN для клиентского приложения запросившего авторизацию. В случае не авторизации пользователя формируется ответ с ошибкой авторизации пользователя.
4. При успешной авторизации пользователя клиентскому приложению передается выданный ACCESSTOKEN и URL для получения RESOURCETOKEN и REFRESHTOKEN на следующем этапе. В случае неудачи авторизации пользователя, клиентскому приложения пересылается данные об ошибке сформированные на предыдущем этапе.
5. Для получения RESOURCETOKEN и REFRESHTOKEN клиентское приложение отправляет свой SECRETTOKEN и полученный ACCESSTOKEN на данное соединение.
6. OAUTH-SERVER после авторизации клиентского приложения по полученным данным отправляет REFRESHTOKEN (это автоматическая замена пере-аутентификации пользователя, которая не требует лишний раз

вводить логин и пароль) и RESOURCETOKEN для последующего обмена данными. В случае если произошла ошибка авторизуемого соединения, формируется объект ошибки и отправляется клиентскому приложению.

Область информационной безопасности является компетенцией отдела программного и технического обеспечения Рубцовского института (филиала) АлтГУ. Данным отделом регулярно производятся плановые профилактические работы всех программно-аппаратных систем Института.

3 Оценка эффективности внедрения информационной системы

3.1 Общие положения

При разработке новой информационной системы для любой организации особенно важен вопрос об эффективности выполняемых работ.

Эффективность ИС представляет собой свойство системы, которое выполняет поставленную цель в заданных условиях использования, и с определенным качеством. Такая характеристика отражает:

- действенность системы;
- техническое совершенство ИС;
- простоту и технологичность разработки и создания системы;
- удобство использования и обслуживания системы;
- улучшение и облегчение условий труда, изменение его содержания, развитие творческих функций, способностей и потребностей людей, преодоление существенных различий в труде и др.;
- экономическую целесообразность внедрения ИС.

Понятие эффективности направленно на получение выгодного результата – эффекта использования.

Задачами, стоящими при создании ИС являются минимизация стоимости и обеспечение требуемого качества ИС.

Свойства системы, обозначающие возможность её использования, для удовлетворения определенных потребностей пользователей в соответствии с ее назначением называются качеством.

К основным показателям качества ИС обычно относят: безопасность, достоверность и надежность.

Безопасность представляет собой свойство, заключающееся в

способности системы обеспечить конфиденциальность и целостность информации.

Достоверность функционирования – свойство системы, обуславливающее безошибочность производимых ею преобразований информации.

Надежность представляет собой средство обеспечения актуальной и достоверной информации на выходе системы.

3.2 Показатели эффективности

Показатели эффективности показывают степень приспособленности системы к выполнению поставленных перед нею задач и являются обобщающими показателями оптимальности функционирования ИС. Кардинальными обобщающими показателями являются показатели экономической эффективности системы, характеризующие целесообразность произведенных на создание и функционирование системы затрат [20].

В качестве показателей экономической эффективности применяют:

- годовой экономический эффект;
- коэффициент экономической эффективности капитальных вложений;
- срок окупаемости капитальных вложений;
- трудоемкость обработки информации;
- эксплуатационная стоимость затрат;
- расчет текущих затрат пользователя;
- экономия текущих затрат при автоматизации;
- годовая экономия затрат на материалы.

Результатом внедрения мероприятия, который выражен в стоимостной форме, называют экономическим эффектом.

Источниками экономии являются:

− рост объёмов и сокращение сроков переработки информации;

− повышение коэффициента использования вычислительных ресурсов;

− снижение затрат на эксплуатационные материалы.

Прежде чем осуществлять внедрение ИС, необходимо дать экономическое обоснование целесообразности внедрения. Экономическая целесообразность означает, что создание ИС должно предусматривать выбор таких проектных решений, которые при условии достижения поставленных целей и задач, обеспечивают минимизацию затрат финансовых, материальных и трудовых ресурсов.

Вместе с экономической эффективностью также можно говорить о прагматической, технической, эксплуатационной и социальной эффективностей.

Среди показателей прагматической эффективности выделяют:

− показатели достоверности преобразования информации;

− показатели безопасности информационной системы;

− показатели точности вычислений и преобразования информации;

− показатели полноты формирования системой результатной информации;

− показатели оперативности.

Используя показатели технической эффективности, оценивается техническое совершенство системы, научно-технический уровень организации и функционирование системы.

В качестве показателей эксплуатационной эффективности могут выступать показатели надежности, функциональность системы, количество обслуживаемых абонентов, производительность, пропускная способность, тактовая частота, временные задержки, емкость памяти, эксплуатационные характеристики и т.п.

Показателями социальной эффективности являются образ и качество жизни, отражающееся в конечном итоге в продолжительности жизни человека и всего населения страны.

3.3 Расчет экономической эффективности

3.3.1 Смета затрат на разработку

Затраты на проектирование и реализацию программы состоят из затрат на покупные материалы, основной заработной платы, дополнительной заработной платы, страховых взносов, накладных расходов, затрат на машинное время (затрат на электроэнергию).

Затраты на покупные изделия рассчитываются по формуле (3.1).

$$З_{мi} = Ц_{едi} * N_{мi}\ , \quad (3.1)$$

где $З_{мi}$ – затраты на i-й материал, $Ц_{едi}$ – цена за единицу i-го материала, $N_{мi}$– количество i-го материала.

Общие затраты на материалы определяются по формуле (3.2).

$$З_{м} = \sum З_{мi}\ , \quad (3.2)$$

где $З_{м}$ – общие затраты на материалы.

В таблице 3.1 приведен перечень затрат на материалы и покупные изделия.

Таблица 3.1 – Затраты на материалы и покупные изделия

№ п/п	Наименование материала	Кол-во, шт.	Цена за единицу, руб.	Стоимость, руб
1	Доступ в Internet	1	750	750

Продолжение таблицы 3.1

2	Канцелярские товары	5	10	50
3	Бумага формата А4	300	0,90	270
4	Итого			1070

Таким образом, общие затраты на материалы составляют 1070 руб.

Далее произведем расчет фонда заработной платы (основной и дополнительной заработной платы разработчика (программиста)) по формуле (3.3).

$$З_{нач} = З_{осн} + З_{доп}, \quad (3.3)$$

где $З_{осн}$– размер основной заработной платы, $З_{доп}$– размер дополнительной заработной платы, в которую входят выплаты, предусмотренные трудовым договором, определяется по формуле (3.4).

$$З_{доп} = З_{осн} * 0{,}1\ , \quad (3.4)$$

Результаты расчета представлены в таблице 3.2.

Таблица 3.2 – Расчет фонда заработной платы

№ п/п	Должность: разработчик (программист)	Кол-во рабочих дней	Кол-во проработанных дней	Размер дневной оплаты, руб.	Заработная плата, руб.
1	Основная заработная плата	47	47	530	24 910
2	Дополнительная заработная плата				2 491
3	Итого фонд заработной платы				27 401

Таким образом, фонд заработной платы разработчика составляет 27 401 руб.

К отчислениям на социальные нужды относят страховые взносы в ПФР, ФСС, ФОМС и взносы на страхование от несчастных случаев на

производстве и профессиональных заболеваний.

Отчисления в пенсионный фонд определяются по формуле (3.5).

$$З_{ПФ} = З_{нач} * 0{,}22 = 27\ 401 * 0{,}22 = 6\ 028{,}22 \text{ руб.}, \quad (3.5)$$

где $З_{ПФ}$ – размер отчислений в пенсионный фонд.

Отчисления на социальное страхование рассчитываются по формуле (3.6).

$$З_{СС} = З_{нач} * 0{,}029 = 27\ 401 * 0{,}029 = 794{,}63 \text{ руб.}, \quad (3.6)$$

где $З_{СС}$ – размер отчислений на социальное страхование.

Отчисления в фонд обязательного медицинского страхования определяются по формуле (3.7).

$$З_{МС} = З_{нач} * 0{,}051 = 27\ 401 * 0{,}051 = 1\ 397{,}45 \text{ руб.}, \quad (3.7)$$

где $З_{МС}$ – размер отчислений в фонд обязательного медицинского страхования.

Отчисления на обязательное социальное страхование от несчастных случаев на производстве рассчитываются по формуле (3.8).

$$З_{НС} = З_{нач} * 0{,}002 = 27\ 401 * 0{,}002 = 54{,}8 \text{ руб.}, \quad (3.8)$$

где $З_{НС}$ – размер отчислений на обязательное страхование от несчастных случае на производстве.

В таблице 3.3 представлены численные значения отчислений на социальные нужды.

Таблица 3.3 – Расчет отчислений на социальные нужды (страховые взносы)

№ п/п	Отчисления на социальные взносы (страховые нужды)	Тарифы страховых взносов, в %	Суммы страховых взносов, руб.
1	ПФР	22,00	6 028,22
2	ФСС	2,90	794,63
3	ФОМС	5,10	1 397,45
4	На обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний	0,20	54,8
5	Итого	30,20	8 275,1

Стоимость машинного времени зависит от себестоимости машино-часа работы машины, времени работы и амортизацию машины и оборудования, а так же затраты на электроэнергию ((3.9) – (3.11)).

$$А_м = \frac{О_ф * Н_{ам}}{365*100} * Т_м \text{ ,} \quad (3.9)$$

Среднестатистическая стоимость компьютера составляет 35000 рублей, норма амортизации, принята равной 25%. Таким образом, $А_м$ = (700000/ 36500)*70 = 1342,5 руб.

Рассчитаем дополнительные расходы к основным затратам на процессы производства и обращения. Накладные расходы $З_н$ фирмы составляют 20 процентов (условно) от суммы основной и дополнительной заработной платы.

$$З_н = З_{нач} * 0,2 = 27\,401 * 0,2 = 5\,480,2 \text{ руб.} \quad (3.10)$$

Рассчитаем затраты на машинное время.

Как следует из данных таблицы 3.1, на разработку и тестирование веб-

приложения потребовалось 47 рабочих дня ($Д_н$).

В среднем с учетом перерывов программист работает за компьютером 7 часов в день. Себестоимость одного кВт/ч электроэнергии ($С_{1квт/ч}$) для организаций составляет 6 рублей 50 копеек.

Складываем мощность энергопотребителей для программиста из мощности, потребляемой системным блоком компьютера, монитором, и другим периферийным оборудованием, которая составляет 1,2 кВт. Следовательно, за 7 часов работы программиста суммарное энергопотребление за день составит: P = 1,2 * 7 = 8.4 кВт/ч.

Таким образом, стоимость машинного времени Змаш, необходимого для разработки ИС составит:

$$З_{маш} = Р * Д_н * С_{\frac{1кВт}{ч}} = 8{,}4\ кВт/ч * 47 * 6{,}5\ руб./кВт/ч = 2\ 566{,}2\ руб. \quad (3.11)$$

Затраты на машинное время учитываются как затраты на электроэнергию.

В результате вышепроизведенных расчетов были получены итоговые затраты на разработку, которые указаны в таблице 3.4.

Таблица 3.4 – Итоговая смета затрат

№ п/п	Наименование статей расхода	Сумма, руб.
1	Стоимость материалов и покупных изделий	1 070
2	Основная заработная плата	24 910
3	Дополнительная заработная плата	2 491
4	Отчисления за социальные нужды	8 275,1
5	Амортизация ЭВМ и оборудования	1 342,5
6	Накладные расходы	5 480,2
7	Затраты на машинное время (затраты на электроэнергию)	2 566,2
8	Итого	46 135

Цена программного продукта определяется по формуле (3.12).

$$Ц = З_{итог} + П = З_{итог} + З_{нач} * 0{,}3 = 46\,135 + 27\,401 * 0{,}3 = 54\,355{,}3, \quad (3.12)$$

где Ц – итоговая цена программного продукта, $З_{итог}$ – итоговые затраты, П – прибыль, которая, составляет 30 процентов (условно) от фонда заработной платы.

3.3.2 Оценка экономической эффективности

Для описания экономической эффективности необходимо сопоставить существующие и внедряемые технологии.

Для организации расчета трудоемкости обработки информации были введены два основных показателя: T_0 – это трудозатраты на обработку информации в базовом варианте (чел/час), T_j – трудозатраты на обработку информации по предлагаемому варианту (чел/час).

Для проектируемого бизнес-процесса следует рассмотреть уже оптимизированные бизнес-процессы, что дает время T_j (таблица 3.5).

Таблица 3.5 – Показатели величины трудоемкости обработки информации

№ п/п	Наименование операции	Базовый вариант (T_0)		Проектный вариант (T_j)	
		Минут за сутки	Часов за год	Минут за сутки	Часов за год
1	Установка новой ИС на рабочие компьютеры сотрудников	30	7	1	0,28
2	Процесс авторизации пользователя в необходимых для работы ИС	1	4	0,08	0,3
3	Разработка новой ИС	-	30	-	5
4	Всего	31	41	1,08	5,58

Показатель снижения трудовых затрат (ΔТ) рассчитывается по формуле (3.13).

$$\Delta T = T_0 - T_j = 41 - 5{,}58 = 35{,}42 \text{ чел/час.} \quad (3.13)$$

Индекс снижения трудовых затрат или повышения производительности труда (Y_T) рассчитывается по формуле (3.14).

$$Y_T = T_0 \div T_j = 41 \div 5{,}58 = 7{,}35. \quad (3.14)$$

Коэффициент относительного снижения трудовых затрат $К_Т$ рассчитывается по формуле (3.15).

$$К_Т = \Delta T / T_0 * 100\% = 35{,}42/41*100 = 86\ \% \quad (3.15)$$

Таким образом, на 86 % снижаются трудовые затраты.

Кроме того, из таблицы 3.5 видно, что невысокие затраты на разработку web-приложения, принесут еще и значительную эффективность от внедрения новых информационных систем. Время на внедрение подприложений уменьшится, так как у родительского приложения имеется шаблон подприложения.

Помимо этого, web-браузер является встроенной системой в большинство операционных систем, что позволяет упростить работу техников Института в установке, настройке, обновлении, а также обслуживании рабочих компьютеров сотрудников Института.

ЗАКЛЮЧЕНИЕ

Целью выпускной квалификационной работы являлась разработка web-приложения интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ.

Для достижения поставленной цели были решены следующие задачи:

– проведен анализ предметной области;

– выявлены недостатки существующей КИС Института;

– выработаны проектные решения по функциональному обеспечению разрабатываемого web-приложения;

– описаны проектные решения по обеспечивающим подсистемам разрабатываемого web-приложения;

– выполнен обзор программных продуктов;

– разработано web-приложение интеграции компонентов корпоративной информационной системы Рубцовского института (филиала) АлтГУ;

– оценена эффективность от внедрения web-приложения.

Внедрение web-приложения позволит достичь следующих положительных сторон:

1. Единая точка идентификации и авторизации.
2. Представление информации при помощи web-браузера.
3. Возможность организации удаленной работы с корпоративной информационной системой Института.
4. Беспрерывная поставка обновлений ИС, а также удобное внедрение новых информационных систем в КИС.
5. Единая система оповещения и информирования сотрудников.

Таким образом, поставленная цель в рамках выпускной квалификационной работы достигнута.

В перспективе, в созданное web-приложение будут интегрированы остальные ИС Рубцовского института (филиала) АлтГУ.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Рубцовский институт (филиал) АлтГУ [Электронный ресурс] / Режим доступа: https://rb.asu.ru/ – Загл. с экрана.

2. Астапчук, В. А. Корпоративные информационные системы: требования при проектировании : учебное пособие для вузов / В. А. Астапчук, П. В. Терещенко. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2022. — 113 с. — (Высшее образование). — ISBN 978-5-534-08546-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/492141 – Загл. с экрана.

3. Chesney, Robert, Cybersecurity Law, Policy, and Institutions (version 3.0) (August 24, 2021). Available at SSRN: https://ssrn.com/abstract=3547103. – Загл. с экрана.

4. Введение в OAuth 2 [Электронный ресурс]. – Режим доступа: https://www.digitalocean.com/community/tutorials/oauth-2-ru – Загл. с экрана.

5. Зенков, А. В. Информационная безопасность и защита информации : учебное пособие для вузов / А. В. Зенков. — Москва : Издательство Юрайт, 2022. — 104 с. — (Высшее образование). — ISBN 978-5-534-14590-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/497002 – Загл. с экрана.

6. Казарин, О. В. Надежность и безопасность программного обеспечения : учебное пособие для вузов / О. В. Казарин, И. Б. Шубинский. — Москва : Издательство Юрайт, 2022. — 342 с. — (Высшее образование). — ISBN 978-5-534-05142-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/493262 – Загл. с экрана.

7. UML Designer – обзор сервиса [Электронный ресурс]. – Режим доступа: https://www.umldesigner.org/ – Загл. с экрана.

8. Micro-zoe [Электронный ресурс]. – Режим доступа: https://micro-zoe.github.io/micro-app/docs.html#/ – Загл. с экрана.

9. Информационные системы в экономике : учебник для вузов / В. Н. Волкова, В. Н. Юрьев, С. В. Широкова, А. В. Логинова ; под редакцией В. Н. Волковой, В. Н. Юрьева. — Москва : Издательство Юрайт, 2022. — 402 с. — (Высшее образование). — ISBN 978-5-9916-1358-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/489695 – Загл. с экрана.

10. Полуэктова, Н. Р. Разработка веб-приложений : учебное пособие для вузов / Н. Р. Полуэктова. — Москва : Издательство Юрайт, 2022. — 204 с. — (Высшее образование). — ISBN 978-5-534-13715-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/496682 – Загл. с экрана.

11. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2022. — 218 с. — (Высшее образование). — ISBN 978-5-534-00515-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/490128 – Загл. с экрана.

12. Организационное и правовое обеспечение информационной безопасности : учебник и практикум для вузов / под редакцией Т. А. Поляковой, А. А. Стрельцова. — Москва : Издательство Юрайт, 2022. — 325 с. — (Высшее образование). — ISBN 978-5-534-03600-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/498844 – Загл. с экрана.

13. Системный анализ : учебник и практикум для вузов / В. В. Кузнецов [и др.] ; под общей редакцией В. В. Кузнецова. — Москва : Издательство Юрайт, 2022. — 270 с. — (Высшее образование). — ISBN 978-5-9916-8591-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/490660 – Загл. с экрана.

14. Зараменских, Е. П. Управление жизненным циклом информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2022. — 497 с. — (Высшее образование). — ISBN 978-5-534-14023-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/489983 – Загл. с экрана.

15. Янцев, В. В. JavaScript. Как писать программы : учебное пособие для вузов / В. В. Янцев. — Санкт-Петербург : Лань, 2022. — 200 с. — ISBN 978-5-8114-8559-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/197547 – Загл. с экрана.

16. Кириченко, А. В. JavaScript для FrontEnd-разработчиков. Написание. Тестирование. Развертывание / А. В. Кириченко. — Санкт-Петербург : Наука и Техника, 2020. — 320 с. — ISBN 978-5-94387-789-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/175389 – Загл. с экрана.

17. Асалханов, П. Г. Web-программирование: JavaScript : учебное пособие / П. Г. Асалханов. — Иркутск : Иркутский ГАУ, 2020. — 123 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/183488 – Загл. с экрана.

18. Волкова, В. Н. Теория информационных процессов и систем : учебник и практикум для вузов / В. Н. Волкова. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 432 с. — (Высшее образование). — ISBN 978-5-534-05621-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/489220 – Загл. с экрана.

19. Богатырев, В. А. Информационные системы и технологии. Теория надежности : учебное пособие для вузов / В. А. Богатырев. — Москва : Издательство Юрайт, 2022. — 318 с. — (Высшее образование). — ISBN 978-5-534-00475-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/490026 – Загл. с экрана.

20. Оценка эффективности внедрения информационной системы управления предприятием. Измеримые цели и контроль их достижения [Электронный ресурс]. – Режим доступа: https://www.topsbi.ru/about-the-company/presscentr/publikacii/ocenka_effektivnosti_vnedreniya_informacionnoy_sistemy_upravleniya_predpriyatiem/. – Загл. с экрана.

21. Внуков, А. А. Защита информации : учебное пособие для вузов / А. А. Внуков. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 161 с. — (Высшее образование). — ISBN 978-5-534-07248-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/490277 – Загл. с экрана.

ПРИЛОЖЕНИЕ А

Диаграмма корпоративной информационной системы Рубцовского института (филиала) АлтГУ «Как есть»

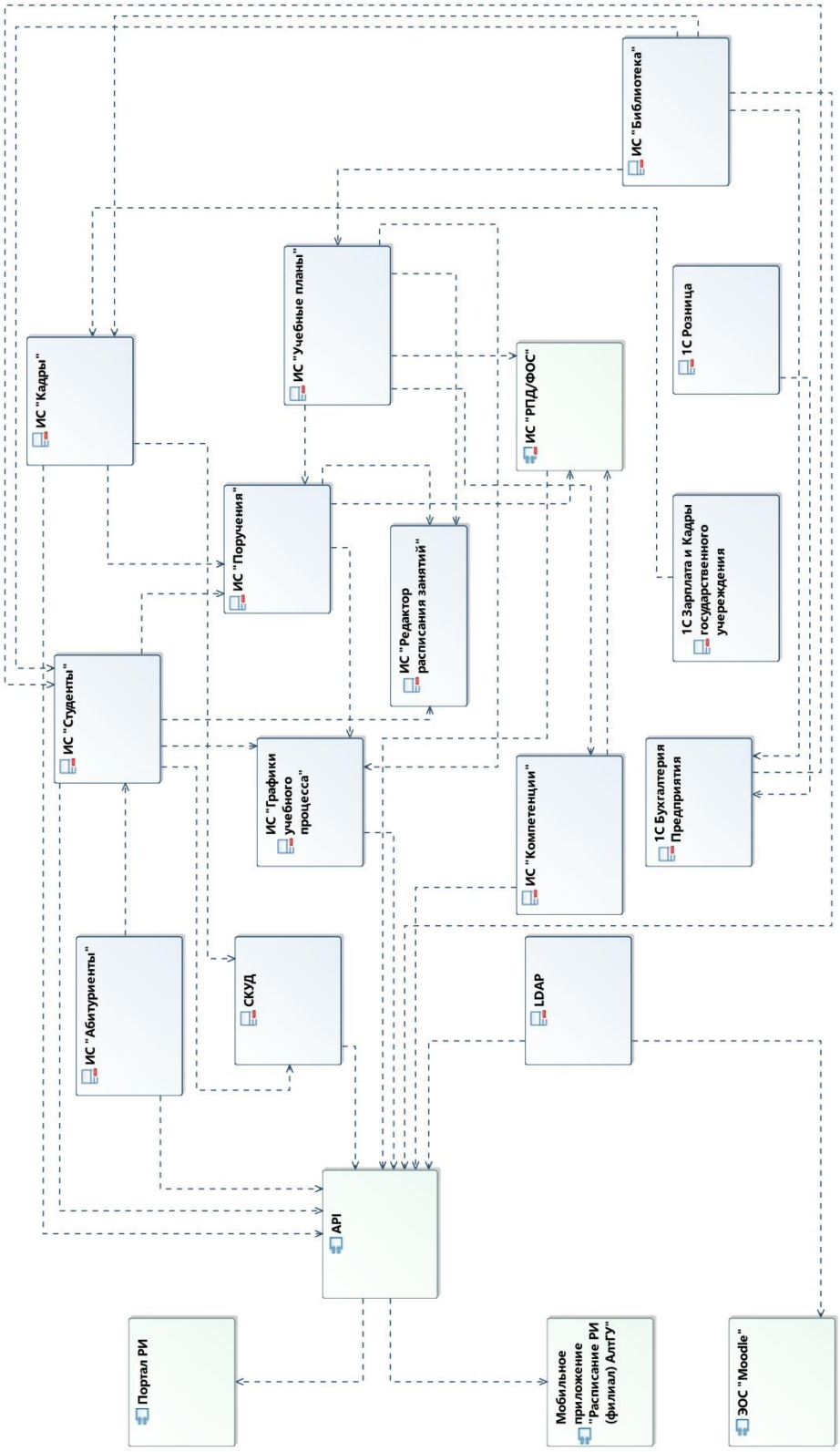


Рисунок А.1 – Компонентная диаграмма «Как есть»

ПРИЛОЖЕНИЕ Б

Диаграмма корпоративной информационной системы Рубцовского института (филиала) АлтГУ «Как должно быть»

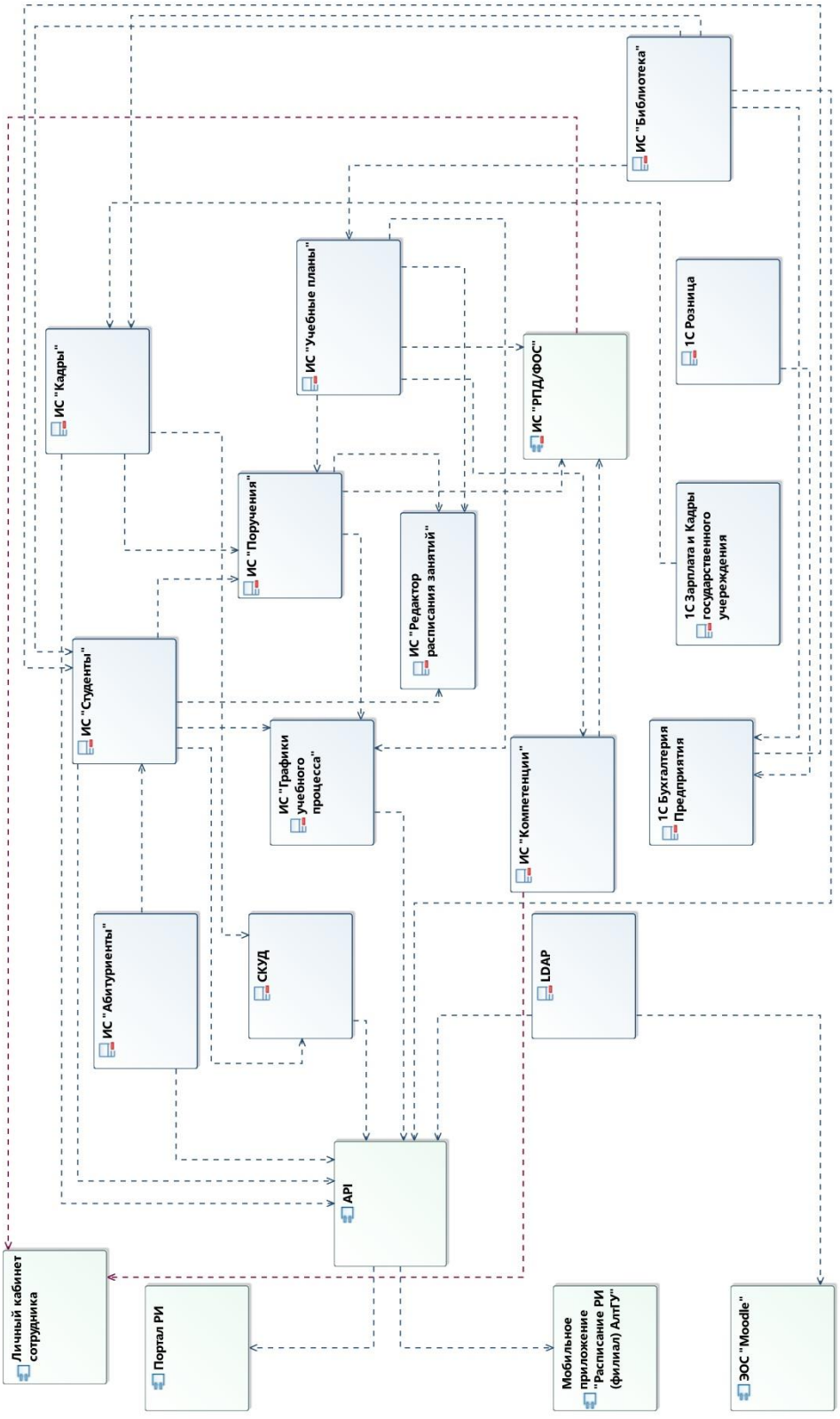


Рисунок Б.1 – Компонентная диаграмма «Как должно быть»