

РЕФЕРАТ

Выпускная квалификационная работа 63 страницы, 3 части, 18 рисунков, 4 таблицы, 20 источников.

Ключевые слова: учебно-методическая документация, рабочие программы дисциплин, фонды оценочных средств, C#, .NET Core, Visual Studio Enterprise 2022, Oracle Database, Swagger, Dapper.

Целью выпускной квалификационной работы является разработка программного интерфейса web-приложения «Информационная система формирования учебно-методической документации для дисциплин» (на примере Рубцовского института (филиала) АлтГУ).

Для достижения поставленной цели необходимо решить ряд задач:

- 1) рассмотреть и проанализировать существующее состояние предметной области, характеристики объекта и системы управления, характеристики используемых информационных систем;
- 2) выявить недостатки функционирования объекта и обосновать предложения по устранению выявленных недостатков;
- 3) описать состав и структуру разрабатываемой системы, связей между функциональными подсистемами;
- 4) описать элементы и модули информационного, технического, программного других обеспечивающих подсистем;
- 5) разработать программный интерфейс web-приложения;
- 6) оценить эффективность внедрения информационной системы.

Объектом исследования данной выпускной квалификационной работы является Рубцовский институт (филиал) федерального государственного образовательного учреждения высшего образования «Алтайский государственный университет».

Предметом исследования выпускной квалификационной работы является процесс формирования компонентов учебно-методической документации для дисциплин.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Аналитическая часть	6
1.1 Технико-экономическая характеристика предметной области.....	6
1.2 Анализ функционирования объекта исследования	13
1.3 Определение цели и задач проектирования ИС.....	17
1.4 Обзор и анализ существующих разработок	17
1.5 Выбор и обоснование проектных решений.....	20
1.5.1 Технологическое обеспечение	20
1.5.2 Программное обеспечение	22
1.5.3 Техническое обеспечение.....	25
1.5.5 Обеспечение информационной безопасности	26
2 Проектная часть	27
2.1 Разработка функционального обеспечения.....	27
2.2 Разработка информационного обеспечения.....	29
2.2.1 Используемые классификаторы и системы кодирования.....	29
2.2.2 Характеристика нормативно-справочной и входной оперативной информации	29
2.2.3 Характеристика результатной информации.....	31
2.2.4 Информационная модель и ее описание.....	33
2.3 Разработка программного обеспечения.....	37
2.3.1 Структурная схема функций управления и обработки данных .	37
2.3.2 Описание программных модулей	40
2.3.3 Схема взаимосвязи программных модулей и информационных файлов	44
2.3.4 Компоненты пользовательского интерфейса.....	45
2.4 Компьютерно-сетевое обеспечение	46
2.5 Обеспечение информационной безопасности	48

3 Оценка эффективности внедрения информационной системы.....	50
3.1 Общие положения.....	50
3.2 Показатели эффективности.....	52
3.3 Расчет экономической эффективности.....	54
ЗАКЛЮЧЕНИЕ	59
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	61

ВВЕДЕНИЕ

Актуальность данной выпускной квалификационной работы обуславливается недостатками в процессе формирования компонентов учебно-методической документации для дисциплин в Рубцовском институте (филиале) АлтГУ:

1. Преподавателям института необходимо каждый год формировать большой объем учебно-методической документации для дисциплин. Это требование федеральных государственных образовательных стандартов. Данная документация в настоящее время формируется вручную. Каждый документ хранится в виде отдельного файла Microsoft Word.

2. Вся справочная информация, необходимая для формирования учебно-методической документации, предоставляется в виде большого количества файлов: стандарты и положения в виде документов Microsoft Word и PDF, учебные планы в виде больших таблиц Microsoft Excel. Кроме этого, преподаватели пользуются документацией, сформированной в прошлые годы. В таком количестве документации бывает сложно ориентироваться, а в электронных таблицах сложно находить информацию для конкретной дисциплины.

3. Так как процесс формирования документов занимает длительно время, преподаватели могут уставать в процессе и из-за этого допускают ошибки в документах: по невнимательности могут указать неверное количество часов, выделенное для дисциплины, неверно указать семестр, в котором будет проходить дисциплина, неправильно указать наименование раздела или подраздела.

Объектом исследования данной выпускной квалификационной работы является Рубцовский институт (филиал) федерального государственного образовательного учреждения высшего образования «Алтайский государственный университет».

Предметом исследования выпускной квалификационной работы является процесс формирования компонентов учебно-методической документации для дисциплин.

Целью выпускной квалификационной работы является разработка программного интерфейса web-приложения «Информационная система формирования учебно-методической документации для дисциплин» (на примере Рубцовского института (филиала) АлтГУ).

Для достижения поставленной цели необходимо решить ряд задач:

- 1) рассмотреть и проанализировать существующее состояние предметной области, характеристики объекта и системы управления, характеристики используемых информационных систем;
- 2) выявить недостатки функционирования объекта и обосновать предложения по устранению выявленных недостатков;
- 3) описать состав и структуру разрабатываемой системы, связей между функциональными подсистемами;
- 4) описать элементы и модули информационного, технического, программного других обеспечивающих подсистем;
- 5) разработать программный интерфейс web-приложения;
- 6) оценить эффективность внедрения информационной системы.

1 Аналитическая часть

1.1 Техничко-экономическая характеристика предметной области

Полное официальное наименование: Рубцовский институт (филиал) федерального государственного образовательного учреждения высшего образования «Алтайский государственный университет».

Сокращенное наименование: Рубцовский институт (филиал) АлтГУ.

Рубцовский институт (филиал) АлтГУ является обособленным структурным подразделением ФГБОУ ВО «Алтайский государственный университет», расположенным в городе Рубцовске Алтайского края [14].

Институт создан приказом Государственного комитета Российской Федерации по высшему образованию (в настоящее время: Министерство науки и высшего образования Российской Федерации) от 22 февраля 1996 г. №324 как филиал Алтайского государственного университет в г. Рубцовске.

Институт не является юридическим лицом и осуществляет функции ФГБОУ ВО «Алтайский государственный университет», в том числе функции представительства, в рамках выданной Университету лицензии на образовательную деятельность.

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.

График учебного процесса: понедельник-пятница 08:00-21:10, суббота 08:00-20:50. График работы специалистов и структурных подразделений: понедельник-четверг 08:00-17:00 (обед 12:00-12:48), пятница 08:00-16:00 (обед 12:00-12:48).

Органами управления Институтом являются общее собрание научно-педагогических работников и обучающихся Института, Ученый совет Института, директор Института, Правления Института (рисунок 1.1).

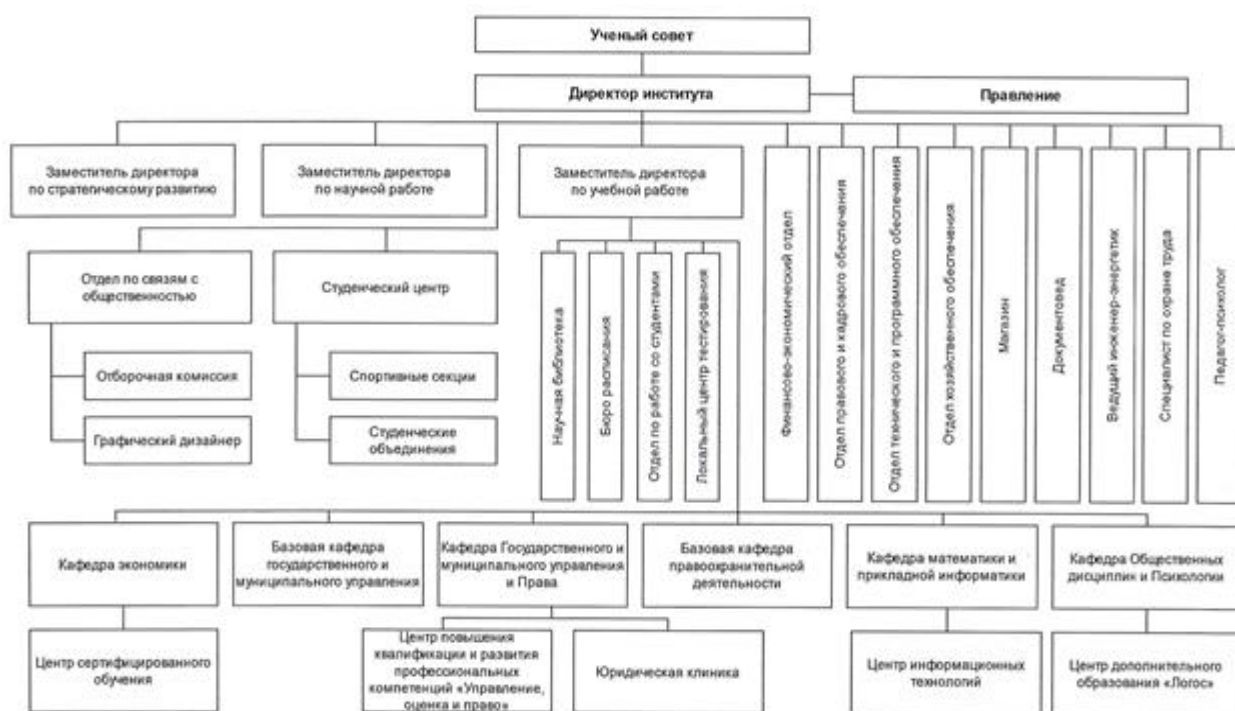


Рисунок 1.1 – Организационная структура управления Рубцовского института (филиала) АлтГУ

Общее собрание научно-педагогических работников и обучающихся Института является коллегиальным органом управления Институтом.

Для общего руководства Институтом создан коллегиальный выборный представительный орган – Ученый совет Института. К компетенции Ученого совета относятся:

- 1) рассмотрение Положения об Институте, предложений о внесении изменений и дополнений к нему и представление соответствующих рекомендаций в Ученый совет Университета;

- 2) утверждение регламента работы Ученого совета;
- 3) утверждение планов работы Ученого совета Института;
- 4) принятие решения о проведении общего собрания научно-педагогических работников и обучающихся Института;
- 5) определение основных перспективных направлений развития Института, рассмотрение программы развития Института;
- 6) рассмотрение плана финансово-хозяйственной деятельности Института и направление его для утверждения ректору Университета;
- 7) утверждение сметы расходов Института на текущий учебный год;
- 8) утверждение принципов распределения штатов между подразделениями Института;
- 9) утверждение структуры Института, принятие решений о создании, реорганизации, ликвидации структурных подразделений Института;
- 10) утверждение положений о структурных подразделениях Института;
- 11) рассмотрение отчетов руководителей структурных подразделений Института;
- 12) принятие решения о создании попечительского совета Института, утверждение его состава, внесение изменений в его состав, а также утверждение регламента его работы;
- 13) проведение конкурсного отбора для избрания на должности профессорско-преподавательского состава, кроме должностей профессора и заведующего кафедрой;
- 14) представление Ученому совету Университета кандидатур из работников Института на присвоение ученых званий, почетных званий, на получение наград Российской Федерации, присуждение премий и др.;
- 15) ежегодное определение на начало учебного года объема учебной нагрузки педагогических работников Института;
- 16) рассмотрение вопросов о представлении работников Института к наградам муниципального уровня;

17) оценка деятельности администрации Института, кафедр и служб на основе отчетов директора, заместителей директора, руководителей подразделений Института;

18) рассмотрение по результатам отчета директора Института и при его согласии вопроса о его рекомендации для заключения трудового договора на новый срок;

19) рассмотрение учебных планов по всем специальностям и направление их для утверждения ректору Университета;

20) утверждение индивидуальных учебных планов образовательных программ, в том числе реализуемых в рамках ускоренной формы обучения;

21) утверждение программ государственной итоговой аттестации, программ практик;

22) решение иных вопросов в соответствии с положением Института.

Непосредственное управление и текущее руководство деятельностью Института, за исключением вопросов, отнесенных действующим законодательством, Уставом Университета или Положением об Институте к компетенции учредителя Университета, его ректора, Ученого совета Университета, иных органов Университета и Института, осуществляет директор.

Для обсуждения основных вопросов организации учебной, учебно-методической, научной, воспитательной, инновационной, хозяйственной деятельности, деятельности по оказания платных услуг и иной приносящей доход деятельности Института создается Правление Института, в состав которого входят директор Института, его заместители, руководители структурных подразделений Института, юрисконсульт.

В таблице 1.1 представлена информация о реализуемых Институтом образовательных программах высшего образования (бакалавриат), формах обучения, нормативных сроках обучения.

Таблица 1.1 – Реализуемые образовательные программы высшего образования (бакалавриат)

Код	Наименование профессии, специальности, направления подготовки	Форма обучения	Срок получения образования
38.03.04	Государственное и муниципальное управление	очно-заочная	5 лет
		очная	4 года
		заочная	4 года 11 месяцев
			5 лет
38.03.02	Менеджмент	очная	не реализуется
		заочная	не реализуется
		очно-заочная	не реализуется
09.03.03	Прикладная информатика	заочная	4 года 11 месяцев
			5 лет
		очная	4 года
37.03.01	Психология	заочная	5 лет
			4 года 11 месяцев
		очно-заочная	5 лет
		очная	4 года
38.03.01	Экономика	очно-заочная	5 лет
		заочная	4 года 11 месяцев
			5 лет
		очная	4 года
40.03.01	Юриспруденция	очная	4 года
		очно-заочная	5 лет
		заочная	4 года 11 месяцев
			5 лет

В таблице 1.2 представлена информация о реализуемых Институтom образовательных программах среднего профессионального образования, формах обучения, нормативных сроках обучения.

Таблица 1.2 – Реализуемые образовательные программы среднего профессионального образования

Код	Наименование профессии, специальности, направления подготовки	Форма обучения	Срок получения образования
21.02.05	Земельно-имущественные отношения	очная	1 год 10 месяцев
			2 года 10 месяцев
09.02.07	Информационные системы и программирование	очная	3 года 10 месяцев
			2 года 10 месяцев
09.02.04	Информационные системы (по отраслям) (в экономике)	очная	3 года 10 месяцев
	Информационные системы (по отраслям) (графический дизайн)		2 года 10 месяцев
40.02.02	Правоохранительная деятельность	очная	3 года 6 месяцев
			2 года 6 месяцев
38.02.06	Финансы	очная	1 год 10 месяцев
			2 года 10 месяцев
38.02.01	Экономика и бухгалтерский учет (по отраслям) (в промышленности, бюджетных отраслях)	очная	1 год 10 месяцев
			2 года 10 месяцев

Отдел технического и программного обеспечения Рубцовского института (филиала) АлтГУ является самостоятельным структурным подразделением и подчиняется непосредственно директору Института.

Задачи отдела технического и программного обеспечения:

- 1) анализ, внедрение и сопровождение новых методов использования локальных и глобальных вычислительных сетей;
- 2) определение возможности использования готовых программных продуктов;

3) обеспечение контроля над правильным использованием оборудования, разработкой инструкций и обучением персонала методов работы с вычислительной техникой, осуществление профилактических и ремонтных работ;

4) своевременное обеспечение электронной техники запасными частями и материалами;

5) надзор над техническим состоянием средств охранной сигнализации.

Функции, выполняемые сотрудниками отдела технического и программного обеспечения:

- организация и проведение работ по разработке, отладке, опытной эксплуатации и поэтапному внедрению комплексов программных и технических средств;

- сопровождение внедренных программ и программных средств;

- разработка и внедрение мероприятий по защите ЛВС от попыток несанкционированного доступа;

- проведение анализа причин отказов и нарушений в работе ЛВС, разработка предложений по их устранению, предупреждению и повышению качества и надежности работы ЛВС Института;

- обеспечение доступа сотрудников Института к ресурсам глобальной сети Интернет;

- методическая помощь различным подразделениям Института в части информационной поддержки;

- участие в работе по совершенствованию документооборота и информационного обмена;

- организация и координация работ по созданию и ведению баз данных;

- организация технического обслуживания, профилактического и текущего ремонта электронной техники;

- изучение возможностей подключения дополнительных внешних устройств к электронно-вычислительным машинам с целью расширения их технических возможностей.

Структуру и штаты отдела технического и программного обеспечения утверждает директор Института в соответствии с нормативами численности специалистов и служащих с учетом объемов работы и особенностей деятельности отдела.

В отделе технического и программного обеспечения оборудованы рабочие места с персональными компьютерами для системного администратора, программистов, техников и др. сотрудников подразделения. Характеристики установленных персональных компьютеров:

- процессор AMD Ryzen 5 3400G with Radeon Vega 11 Graphics (12 нм, 3.7 МГц – 4.2 МГц, 4 ядра, 8 потоков, встроенное графическое ядро Radeon RX Vega 11 Graphics);
- 16 Гб оперативной памяти;
- SSD накопитель 256 Гб;
- HDD накопитель 512 Гб;
- 2 монитора AOC i2475Pхqu (24 дюйма, 60 Гц, IPS, 1920x1080).

1.2 Анализ функционирования объекта исследования

Основными видами деятельности Рубцовского института (филиала) АлтГУ в соответствии с Положением о Институте 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.

На рисунке 1.2 представлена визуальная модель деятельности Рубцовского института (филиала) АлтГУ в нотации функционального моделирования IDEF0.

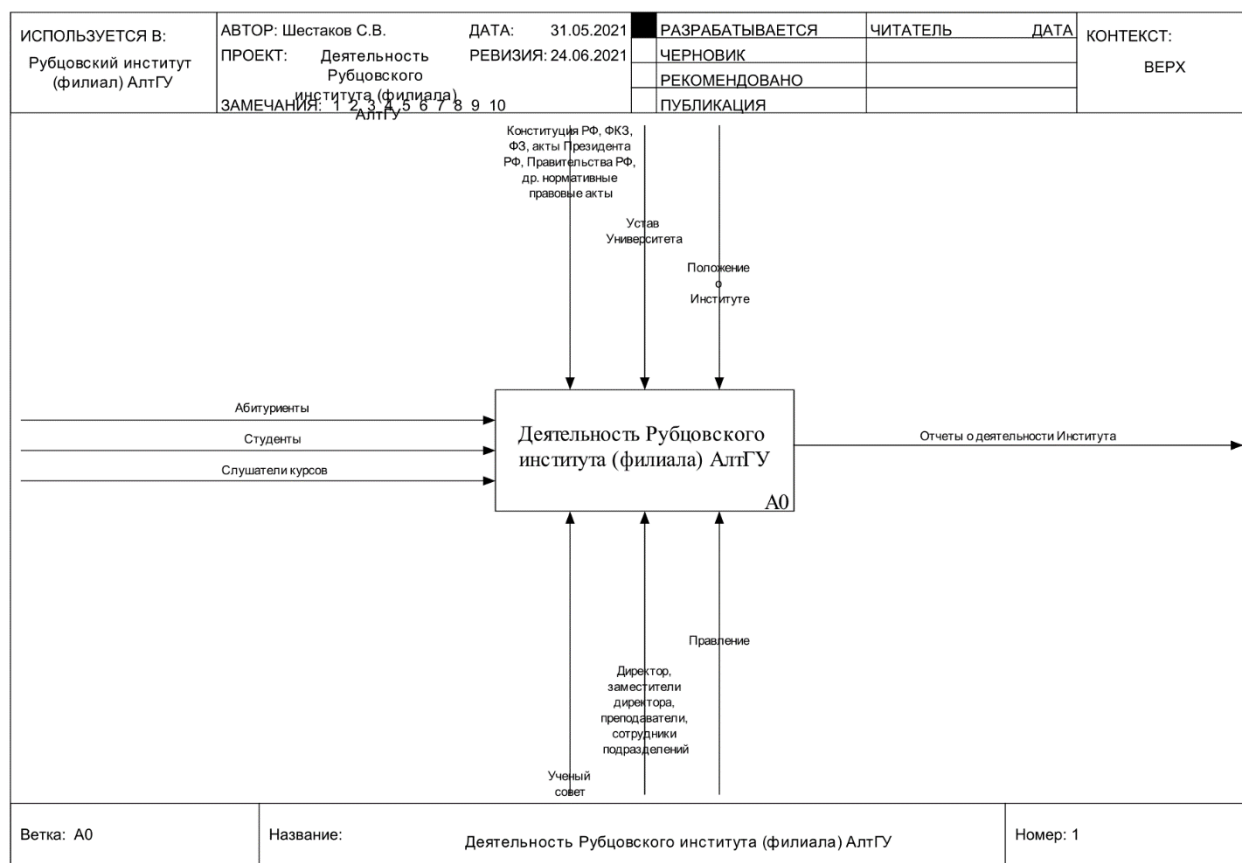


Рисунок 1.2 – Описание деятельности Рубцовского института (филиала) АлтГУ в нотации функционального моделирования IDEF0

Основные процессы деятельности Института выполняются Ученым советом Института, директором, заместителями директора, преподавателями и сотрудниками Института, правлением Института.

На рисунке 1.3 представлена декомпозиция деятельности Рубцовского института (филиала) АлтГУ в нотации функционального моделирования IDEF0.

Для обеспечения основных направлений деятельности Рубцовского института в учебных аудиториях установлено проекционное оборудование

(проекторы, экраны для проекторов), имеется 3 аудитории с мобильными компьютерами и 5 аудиторий со стационарными персональными компьютерами, и читальный зал библиотеки, со стационарными персональными компьютерами.

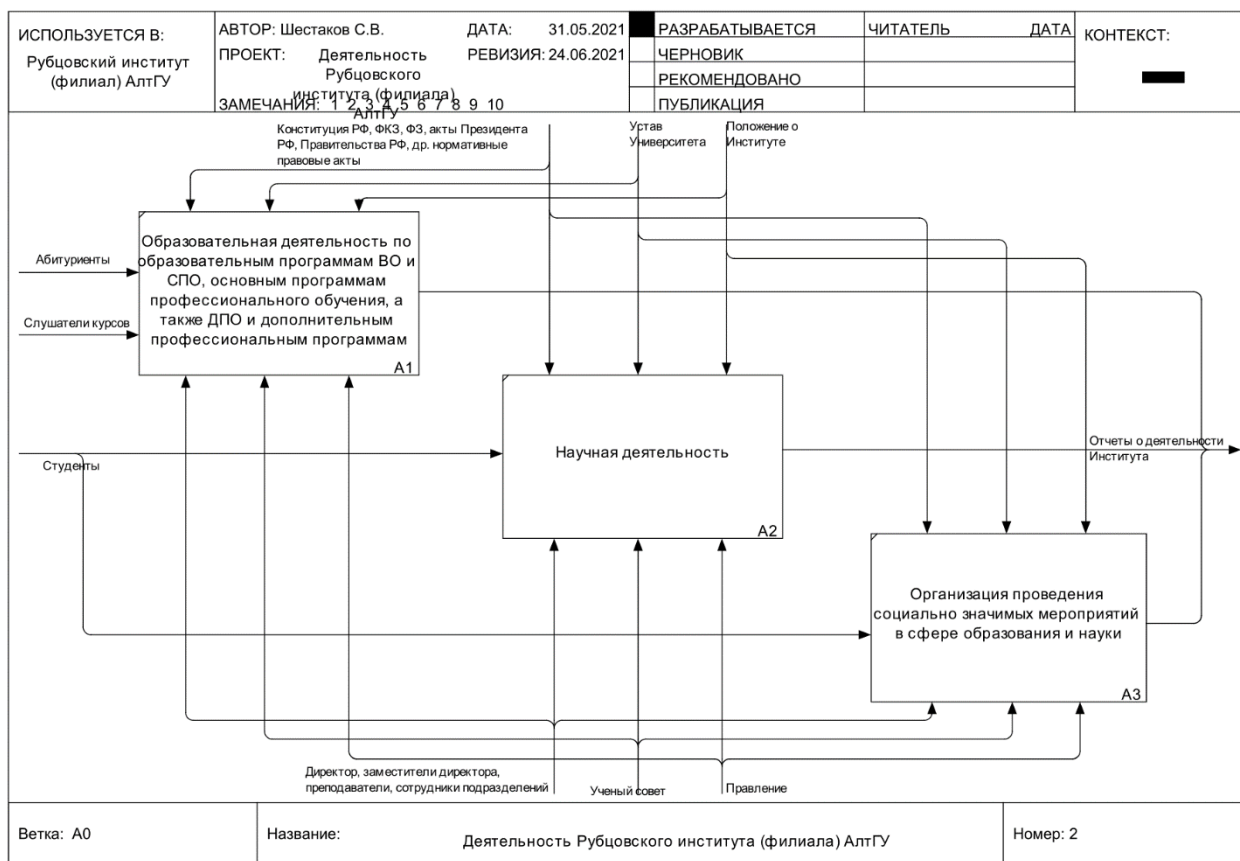


Рисунок 1.3 – Декомпозиция описания деятельности Рубцовского института (филиала) АлтГУ в нотации функционального моделирования IDEF0

Также на каждой кафедре имеются мобильные компьютеры, которые могут быть использованы преподавателями для демонстрации учебного медиаконтента на проекционном оборудовании. Кроме этого, на кафедрах института оборудованы рабочие места со стационарными персональными компьютерами для преподавателей. Данные персональные компьютеры подключены к локальной сети института, с них у преподавателей есть доступ к информационным системам института, а также к сети Интернет. Примерные характеристики персональных компьютеров, установленных на кафедрах института:

- материнская плата MSI G41M-P28 или MSI H110M PRO-VD PLUS;
- процессор Intel Pentium E5500 (45 нм, 2.8 ГГц, 2 ядра) или Intel Pentium G4520 (14 нм, 3.6 ГГц, 2 ядра, 2 потока);
- 2 ГБ оперативной памяти DDR3 или 4 ГБ оперативной памяти DDR4;
- HDD накопитель 80 ГБ или SSD накопитель 120 ГБ.

В институте установлены точки доступа Wi-Fi, доступные для мобильных компьютеров кафедр и учебных аудиторий. Также есть точки доступа Wi-Fi, доступные для личных устройств студентов и сотрудников института. Доступ к Wi-Fi предоставляется по логину и паролю от учетной записи в сети института.

В соответствии с федеральными государственными образовательными стандартами, для каждой дисциплины направления формируется учебно-методическая документация – рабочие программы дисциплин, оценочных средств, к рабочей программе дисциплины [19].

В рабочей программе дисциплины определяются:

- цели освоения дисциплины;
- компетенции, формируемые в результате освоения дисциплины;
- структура и содержание дисциплины (разделы и темы, количество часов для каждой темы, компетенции, формируемые в рамках темы);
- учебно-методическое и информационное обеспечение дисциплины (литература и другие источники, в т.ч. ресурсы сети Интернет);
- материально-техническое обеспечение дисциплины;
- методические указания для обучающихся по освоению дисциплины.

В фонде оценочных средств определяются типовые оценочные средства, необходимые для оценки планируемых результатов обучения по дисциплине и критерии оценивания.

1.3 Определение цели и задач проектирования ИС

Основной целью разработки программного интерфейса web-приложения «Информационная система формирования учебно-методической документации для дисциплин» является устранение недостатков в процессе формирования компонентов учебно-методической документации для дисциплин.

В разрабатываемом программном интерфейсе web-приложения необходимо реализовать следующие функции:

- функция для получения информации о дисциплине из базы данных информационной системы «Учебные планы» (общее количество часов, контрольные точки, профили, форму и сроки обучения по специальности);
- функция для создания компетенций, индикаторов достижения компетенций и прикрепления к компетенциям дисциплин, распределения компетенций и часов по структуре дисциплины;
- функция для создания целей дисциплины, указания знаний, умений и навыков, получаемых в результате освоения дисциплины;
- функция для заполнения справочника источников, учебных лабораторий, программного обеспечения, шаблонов текста описания материально-технического обеспечения и методических указаний;
- функции для формирования списка заданий для фонда оценочных средств.

1.4 Обзор и анализ существующих разработок

В настоящее время на рынке программных средств, позволяющих автоматизировать процесс разработки учебно-методической документации, представлены следующие продукты [10].

Подсистема разработки рабочих программ дисциплин из программного продукта 1С: Университет ПРОФ. Данная подсистема позволяет:

- автоматизировать перенос необходимых данных из учебного плана, ФГОС и профессионального стандарта в рабочие программы дисциплин и автоматически формировать данные для первоначального заполнения рабочих программ;
- разрабатывать новые рабочие программы на основе существующих;
- формировать отчетную документацию о готовности рабочих программ по образовательным программам и учебному плану;
- распечатывать рабочие программы и сводные данные по рабочим программам.

Интеллект Инфо: Образовательные программы является расширением для конфигурации 1С: Университет ПРОФ. Данное расширение предоставляет автоматизацию ряда функций:

- формирование описания учебно-методического и материально-технического обеспечения дисциплины;
- формирование шаблонов рабочих программ дисциплин с автоматическим заполнением первоначальных данных;
- автоматизация процесса утверждения рабочих программ ответственными лицами.

Программный комплекс «Планы» от компании «Лаборатория ММИС». Комплекс обеспечивает автоматизацию документооборота учебного процесса образовательного учреждения и позволяет [9]:

- разрабатывать шаблоны рабочих программ дисциплин на базе учебного плана;
- создавать новые рабочие программы дисциплин на основе существующих;
- контролировать процесс разработки рабочих программ дисциплин;

- импортировать перечень литературы из файлов данных.

Перечисленные программные продукты имеют определенные недостатки, делающие невозможным или сильно усложняющими процесс интеграции данных программных продуктов в информационную среду Рубцовского института:

1. Высокая стоимость лицензии на использование программных продуктов. Стоимость на продукты компании 1С обычно варьируется от 100 до 200 тысяч рублей за программный продукт. Стоимость программного комплекса «Планы» составляет около 100 тысяч рублей, при этом допускается использование программ на неограниченном количестве компьютеров.

2. Для внедрения перечисленных программ в информационную среду института потребуется приобрести внедрить другие программные продукты, так как синхронизировать программные продукты с используемыми в институте информационными системами и базами данных не представляется возможным.

3. Для развертывания всех программных комплексов может потребоваться новое серверное оборудование: продукты 1С как правило очень требовательны к аппаратному обеспечению и сетевой инфраструктуре предприятия. Стоимость нового серверного и сетевого оборудования может достигнуть нескольких миллионов рублей.

Исходя из описания возможностей программных продуктов и их общих недостатков можно сделать вывод, что несмотря на то, что данные продукты позволяют практически полностью решить вопрос автоматизации процесса формирования учебно-методической документации, такие недостатки как высокая стоимость самих продуктов и высокая стоимость оборудования, невозможность синхронизации с информационными системами института, делают неэффективным использование данных программных продуктов. Для решения вопроса автоматизации процесса формирования учебно-методической документации гораздо эффективнее будет разработать такую систему самостоятельно.

1.5 Выбор и обоснование проектных решений

1.5.1 Технологическое обеспечение

Для формирования учебно-методической документации дисциплин, преподаватели используют следующее программное обеспечение:

- операционная система Microsoft Windows 7 или Microsoft Windows 10;
- web-браузер Google Chrome, Mozilla Firefox или Microsoft Edge;
- пакет офисных приложений Microsoft Office (Microsoft Word, Microsoft Excel).

На рисунке 1.4 представлена диаграмма потоков данных (DFD) процесса формирования учебно-методической документации дисциплины «как есть».

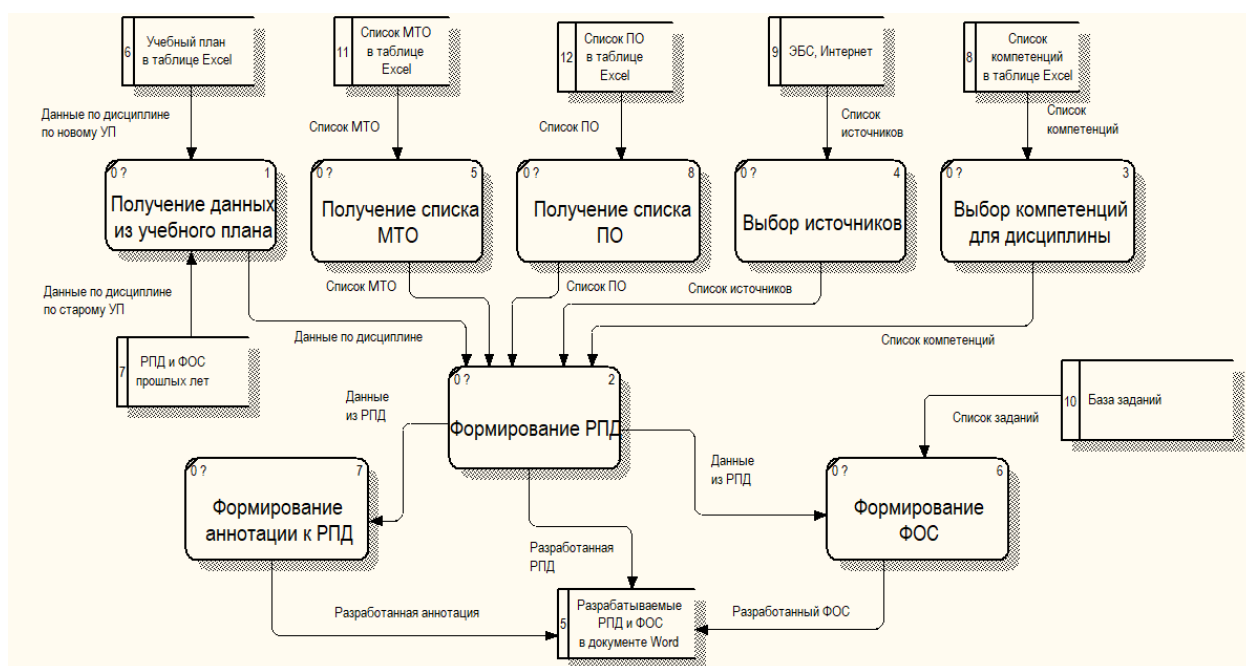


Рисунок 1.4 – Диаграмма потоков данных процесса формирования учебно-методической документации дисциплины «как есть»

Как видно на диаграмме потоков данных, для формирования учебно-методической документации преподавателям необходимо обращаться к большому количеству различных документов и таблиц. В первую очередь для

разработки рабочей программы дисциплины, преподаватели обращаются к электронной таблице Excel, в которой указана информация обо всех дисциплинах для конкретного плана набора. На рисунке 1.5 представлен скриншот электронной таблицы учебного плана для специальности «Информационные системы и программирование». Необходимо обратить внимание, что в таблице представлены данные для всех дисциплин учебного плана, в тот момент, когда преподавателю могут быть необходимы данные только одной конкретной дисциплины.

The screenshot displays an Excel spreadsheet titled '09.02.07 ИСИП на базе 9 кл. 2021_2022.xls (Режим совместимости)'. The spreadsheet is a detailed academic plan for the specialty 'Information Systems and Programming'. It is organized into columns for semesters (Семестр 1, 2, 3) and rows for various disciplines and modules. The table includes fields for credits (кредиты), ECTS (ЕКТС), and other academic metrics. The interface shows standard Excel menus and a formula bar at the top.

Рисунок 1.5 – Учебный план специальности «Информационные системы и программирование» в электронной таблице Excel

В таких же электронных таблицах хранятся списки компетенций, списки программного обеспечения, которое установлено в институте и может использоваться в образовательном процессе, списки материально-технического обеспечения и др. Источники информации для рабочих программ преподаватели ищут в электронных библиотечных системах, а также на других ресурсах сети Интернет.

Из всех полученных данных формируются рабочие программы дисциплин, на основе которых также формируются: фонд оценочных средств

и аннотация к рабочей программе дисциплин. Для разработки фондов оценочных средств также используется база заданий. Преподаватели самостоятельно находят и выбирают базу заданий.

Все документы формируются в файлах Microsoft Word и хранятся на переносных цифровых хранилищах данных.

Использование данного технологического обеспечения приводит к ряду недостатков в процессе формирования учебно-методической документации:

- входная информация размещена в большом количестве документов, что сильно усложняет процесс поиска и анализа данных;
- из-за большого количества избыточных данных в таблицах Excel, преподаватели могут путаться и вносить в разрабатываемые рабочие программы некорректные данные;
- документация формируется вручную и при изменении каких-либо данных необходимо вносить изменения во все файлы вручную. Из-за этого в документах могут появляться ошибки и несостыковки.

1.5.2 Программное обеспечение

На сервере установлена операционная система Microsoft Windows Server 2022. С помощью системы аппаратной визуализации Hyper-V выделяются виртуальные машины для сервера базы данных и сервера API института. На виртуальных машинах используется операционная система Gentoo Linux. На сервере API установлена платформа .NET Core 3.1 для запуска микросервисов, и менеджер процессов PM2.

На персональном компьютере, размещенном на рабочем месте разработчика, установлена операционная система Windows 10 Pro для образовательных учреждений. Лицензия на операционную систему предоставляется Рубцовскому институту по подписке на программное обеспечение от компании Microsoft для образовательных учреждений.

Для описания моделей в нотации IDEF0 и DFD будет использоваться программный продукт Ramus. Это приложение позволяет проводить описание, анализ и моделирование бизнес-процессов, а также строить систему классификации и кодирования. Поддерживает стандартную методологию функционального моделирования IDEF0 и DFD. Позволяет связывать с деятельностью компании практически любые объекты, таким образом реализуя системное сохранение знаний о предприятии. Ramus распространяется по лицензии GNU General Public License v3.0.

Для создания моделей «сущность-связь» базы данных будет использоваться web-приложение draw.io. Draw.io позволяет создавать различные диаграммы и схемы в режиме онлайн, экспортировать проект в изображение формата png, jpeg и т.д. Web-приложение доступно по адресу: <https://draw.io/>.

Работа с базой данных будет организована через объектно-реляционную СУБД Oracle Database от компании Oracle. Данная система управления базами данных используется во всех информационных системах, эксплуатируемых в Рубцовском институте. Использование Oracle Database позволит реализовать интеграцию разрабатываемого API с эксплуатируемыми в Рубцовском институте информационными системами, а также сэкономить ресурсы, которые потребовались бы для развертывания другой СУБД.

Разработка будет производиться в интегрированной среде разработки Visual Studio Enterprise 2022. Данная IDE имеет ряд встроенных инструментов, помогающих разработчику более эффективно разрабатывать и отлаживать программные решения [4]. Для примера, встроенный IntelliCode с помощью искусственного интеллекта обеспечивает контекстно зависимое завершение кода, помогает придерживаться шаблонов и стилей написания кода. Выпуск Visual Studio Enterprise требует наличие лицензии на данное программное обеспечение. У отдела технического и программного обеспечения Рубцовского института есть лицензия на неограниченное количество устройств по подписке на программное обеспечение от компании Microsoft

для образовательных учреждений. Выпуск Enterprise отличается от остальных наличием дополнительных встроенных инструментов для отладки разрабатываемых приложений.

Также для разработки выбран объектно-ориентированный язык программирования C#. C# разработан в 1998-2001 годах группой инженеров компании Microsoft [13, 15]. C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык программирования имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов, делегаты, атрибуты, события, переменные, свойства, обобщенные типы и методы, итераторы, анонимные функции, LINQ, исключения и комментарии в формате XML.

API будет разрабатываться на платформе .NET Core 3.1. Выбор данной платформы для разработки обосновывается тем, что на сервере API Рубцовского института установлена данная платформа, а также запущены и эксплуатируются микросервисы, разработанные на данной платформе. Для примера, список преподавательского состава, размещенный на web-портале Рубцовского института, функционирует с помощью одного из микросервисов, разработанных для платформы .NET Core 3.1.

Для взаимодействия с базой данных из кода разрабатываемого приложения будет использоваться поставщик данных Oracle для платформы .NET – Oracle Data Provider for .NET (ODP.NET) Entity Framework Core и инструмент сопоставления результатов SQL-запросов с классами на языке C# – Dapper [2, 3]. Использование данных инструментов обеспечит оптимизированный доступ к базе данных и высокую производительность при обработке запросов к базе данных.

Документирование разрабатываемого приложения будет происходить автоматически с помощью фреймворка Swagger [1]. Фреймворк автоматически генерирует файлы документации на основе комментариев, оставляемых в коде разработчиком, а также на основе анализа модулей проекта.

Система контроля версий Git и приложение GitHub Desktop. Использование системы контроля версий позволит контролировать процесс разработки приложения, в случае возникновения критических ошибок откатиться до ближайшей исправной версии. Также системы контроля версий упрощают работу с проектом в команде. GitHub является облачным хранилищем файлов проектов. Приложение GitHub Desktop предоставляет графический интерфейс для удобной работы с Git и полную синхронизацию с репозиториями на GitHub.

1.5.3 Техническое обеспечение

База данных и API Рубцовского института размещаются на одном физическом сервере со следующими характеристиками:

- 2 процессора AMD EPYC 7281 (14 нм, 2.1 ГГц или до 2.7 ГГц в режиме Max Boost, 16 ядер, 32 потока);
- 128 Гб оперативной памяти (16 планок по 8 Гб, DDR4, 2666 МГц);
- 2 SSD накопителя Intel D3-S4510 (по 480 Гб каждый);
- 2 SSD накопителя Intel D3-4610 (по 240 Гб каждый);
- 4 HDD накопителя (по 2 Тб каждый);
- 4 HDD накопителя (по 6 Тб каждый);
- RAID адаптер Adaptec RAID 8805 (12 Гбит/с).

На сервере установлена операционная система Microsoft Windows Server 2022. Распределение ресурсов для базы данных и сервера API производится за счет выделения процессов на виртуальные машины с помощью системы аппаратной визуализации Hyper-V.

Характеристики виртуальной машины для базы данных:

- количество виртуальных процессоров – 2;
- 4 Гб оперативной памяти;
- 60 Гб постоянной памяти.

Характеристики виртуальной машины для сервера API:

- количество виртуальных процессоров – 24;
- 24 Гб оперативной памяти;
- 280 Гб постоянной памяти.

На рабочем месте разработчика установлен персональный компьютер со следующими характеристиками:

- процессор AMD Ryzen 5 3400G with Radeon Vega 11 Graphics (12 нм, 3.7 МГц – 4.2 МГц, 4 ядра, 8 потоков, встроенное графическое ядро Radeon RX Vega 11 Graphics);
- 16 Гб оперативной памяти;
- SSD накопитель 256 Гб;
- HDD накопитель 512 Гб;
- 2 монитора AOC i2475Pxqu (24 дюйма, 60 Гц, IPS, 1920x1080).

1.5.5 Обеспечение информационной безопасности

С целью выполнения требований, установленных федеральным законом от 20 июля 2006 года № 152-ФЗ «О персональных данных» в Рубцовском институте было принято «Положение об обработке персональных данных сотрудников Рубцовского института (филиала) АлтГУ» [12]. В соответствии с федеральным законом и данным положением, отделом кадрового и правового обеспечения, и отделом технического и программного обеспечения принимаются меры, направленные на обеспечение выполнения обязанностей, предусмотренных федеральным законом, а также меры по обеспечению безопасности персональных данных при их обработке.

2 Проектная часть

2.1 Разработка функционального обеспечения

На рисунке 2.1 представлена диаграмма потоков данных (DFD) процесса формирования учебно-методической документации дисциплины «как будет». На диаграмме показано, что с внедрением web-приложения «Информационная система формирования учебно-методической документации для дисциплин», для которого разрабатывается программный интерфейс, вся информация будет храниться в единой базе данных на сервере института.

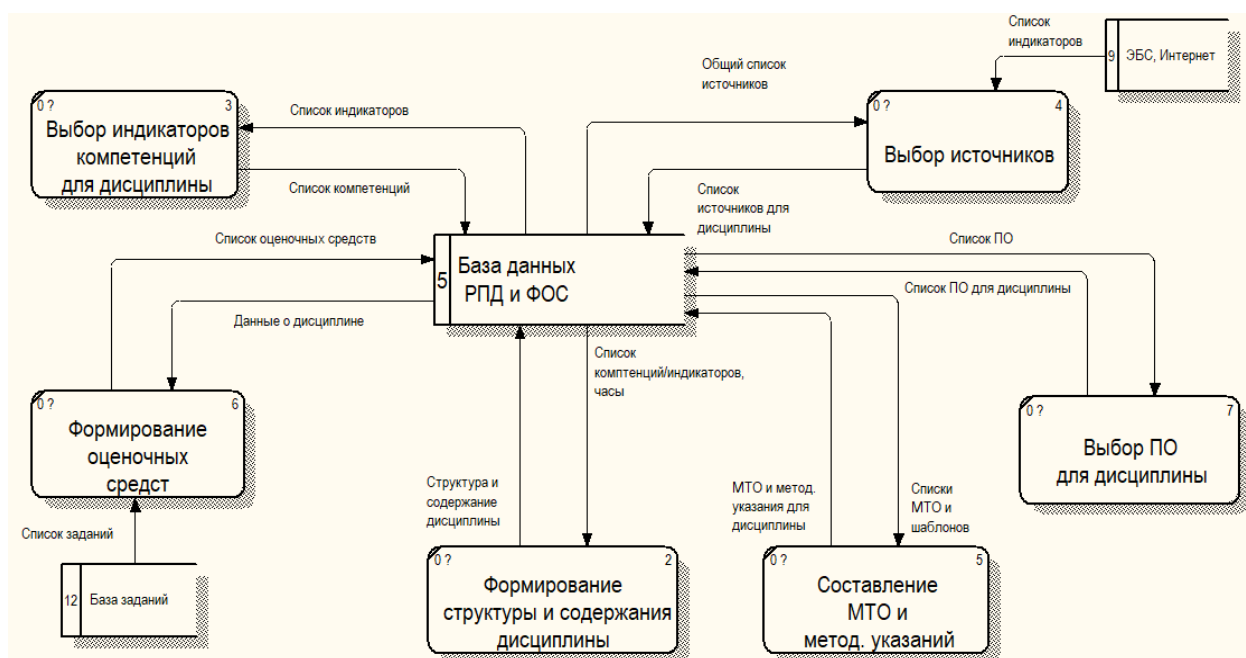


Рисунок 2.1 – Диаграмма потоков данных процесса формирования учебно-методической документации дисциплины «как будет»

На диаграмме потоков данных видно, что теперь вся информация, необходимая для формирования учебно-методической документации, получается не из отдельных документов Microsoft Word и таблиц Microsoft Excel, а из общей базы данных на сервере института.

Для разработки рабочей программы дисциплины преподавателям больше не требуется обращаться к таблице учебных планов и рабочим

программах за прошлые года, чтобы сравнить количество часов, выделенных на дисциплину и семестр, в котором будет проводиться дисциплина. Эти данные автоматически будут сгруппированы в таблице базы данных при первом запросе на получение базовой информации о дисциплине.

Чтобы указать компетенции, реализуемые дисциплиной, также больше не требуется обращаться к таблице Microsoft Excel. Список компетенций заранее будет загружаться в справочник компетенций и методистами распределяться по дисциплинам. От преподавателя потребуются только сформировать структуру и содержание дисциплины, а после распределить список закрепленных за дисциплиной компетенций между темами.

Списки программного обеспечения, учебных лабораторий и материально-технического обеспечения также будут доступны из справочников.

Для закрепления источников информации за рабочей программой дисциплины также доступен справочник источников. Преподавателям доступен открытый справочник источников, в который каждый преподаватель может вносить литературу либо другие источники, после чего закреплять их за рабочей программой дисциплины.

Исходя из информации, описанной на диаграмме потоков данных, можно сделать вывод, что внедрение web-приложения «Информационная система формирования учебно-методической документации для дисциплин» в процесс формирования учебно-методической документации позволит устранить текущие недостатки:

- все необходимые для разработки учебно-методической документации данные будут доступны из базы данных;
- обработка информации будет производиться автоматически сервером, что позволит сократить количество допускаемых ошибок;
- все данные будут доступны в окне единого web-приложения, а не нескольких отдельных ресурсов.

2.2 Разработка информационного обеспечения

2.2.1 Используемые классификаторы и системы кодирования

В разрабатываемом приложении используется порядковая система кодирования. В каждой таблице базы данных, в которых применяется кодирование, используются идентификаторы, уникальные для данной таблицы данных. В большинстве таблиц для кодирования используется одно поле, в котором хранится число – код. При добавлении новой записи в таблицу, ей автоматически присваивается новое уникальное значение кода. Код представляет из себя числовое значение, увеличивающееся на 1 для каждой новой записи. После удаления записи, код повторно не используется.

В некоторых таблицах базы данных используются составные идентификаторы – коды в двух полях. В таком случае, значение в каждом поле идентификатора связано со значением кода в другой таблице (внешний ключ).

2.2.2 Характеристика нормативно-справочной и входной оперативной информации

В модуль справочника «Компетенции» web-приложения «Информационная система формирования учебно-методической документации для дисциплин» необходимо вносить список компетенций из электронной таблицы Microsoft Word (рисунок 2.2). Заполнение таблиц справочника производится методистами кафедр института.

Для каждой специальности создается отдельный список компетенций.

Каждая компетенция имеет свой код и описание. Код состоит из названия категории компетенции (универсальные компетенции, общепрофессиональные компетенции, профессиональные компетенции) и

порядкового номера компетенции. Описание представляет из себя текстовое описание компетенции.

Компетенции, вносимые для специальностей на программах высшего образования, также имеют индикаторы достижения компетенций. Индикаторы достижения компетенций также имеют код и описание. Код индикатора связан с компетенцией, к которой он относится: код компетенции в сочетании с порядковым номером индикатора компетенции. Описание представляет из себя текстовое описание индикатора достижения компетенции.

The image displays two side-by-side screenshots of Microsoft Excel spreadsheets. The left spreadsheet, titled 'R2C1', shows a list of competencies (ОК 01, ОК 02) with their descriptions and associated indicators (ОГ, БД, МД, ОП, МД, ОК). The right spreadsheet, titled 'R2C8', shows a detailed matrix of indicators (ОК 01-08) for various competencies, with cells containing codes like ОК 01, ОК 02, etc.

Рисунок 2.2 – Список компетенций в таблице Microsoft Excel

Компетенции распределяются между дисциплинами данной специальности. По требованиям стандартов, все компетенции должны быть распределены.

Также в разрабатываемой системе используются данные из базы данных информационной системы «Учебные планы». В информационную систему «Учебные планы» каждый год методическим отделом института вносится информация обо всех специальностях и дисциплинах, реализуемых в конкретном учебном году. Из базы данных «Учебных планов» берутся списки

специальностей (коды специальностей, наименования, сроки обучения), список дисциплин для специальности и распределение часов дисциплины по семестрам.

Преподаватели при разработке учебно-методической документации вносят в систему информацию о структуре и содержании дисциплины. Структура представляет из себя набор разделов, в рамках которых описывается содержание – темы занятий и распределение общего количества часов, отведенного на дисциплину, между занятиями.

2.2.3 Характеристика результатной информации

В качестве результатной информации выступает документация, генерируемая фреймворком Swagger. В документации Swagger описаны все используемые в микросервисах контроллеры, конечные точки, по которым можно обратиться к контроллеру, список HTTP-запросов, возможные ответы сервера на запрос и модели данных (рисунки 2.3-2.4) [1, 20].

Competence	
GET	/api/v1/competence/competence/{idspeciality}/{year} Получить список компетенций для направления или отдельной дисциплины (есть query поля)
POST	/api/v1/competence/competence Добавить компетенцию для направления
PATCH	/api/v1/competence/competence/{idcompetence} Изменить код и/или описание компетенции
DELETE	/api/v1/competence/competence/{idcompetence} Удалить компетенцию
Discipline	
GET	/api/v1/competence/discipline/{idcompetence} Получить список дисциплин для компетенции
POST	/api/v1/competence/discipline Закрепить дисциплину за компетенцией
DELETE	/api/v1/competence/discipline/{iddisciplinename}/{idcompetence} Открепить дисциплину от компетенции
Indicator	
GET	/api/v1/competence/indicator/{idcompetence} Получить список индикаторов компетенции
POST	/api/v1/competence/indicator Добавить индикатор для компетенции
PATCH	/api/v1/competence/indicator/{idindicator} Изменить код и/или описание индикатора
DELETE	/api/v1/competence/indicator/{idindicator} Удалить индикатор

Рисунок 2.3 – Документация Swagger со списком контроллеров и HTTP-методов микросервиса «Компетенции»

Code	Description	Links
200	Success text/plain Controls Accept header. Example Value Model <pre>{ "message": "string", "value": [{ "idCompetence": 0, "code": "string", "description": "string" }] }</pre>	No links
400	Bad Request	No links

Рисунок 2.4 – Пример возвращаемых сервером ответов на запрос

Информация из документации Swagger необходима frontend-разработчику, чтобы описать структуру получаемых данных и правильно обработать HTTP-запросы к программному интерфейсу приложения.

Также результатной информацией являются данные, возвращаемые программным интерфейсом в ответ на HTTP-запросы. Программный интерфейс приложения в ответ на HTTP-запрос возвращает данные в формате JSON (рисунок 2.5). JSON это текстовый формат для обмена данными, основанный на языке программирования JavaScript.

Request URL	
https:// /api/v1/competence/competence/602/2022	
Server response	
Code	Details
200	Response body <pre>{ "message": "success", "value": [{ "idCompetence": 171, "code": "ОПК-1", "description": "способен применять естественнонаучные и общинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности" }, { "idCompetence": 189, "code": "ОПК-2", "description": "способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности" }, { "idCompetence": 199, "code": "ОПК-3", "description": "способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований" }] }</pre>

Рисунок 2.5 – Данные, передаваемые сервером в формате JSON

2.2.4 Информационная модель и ее описание

На рисунке 2.6 представлена диаграмма «сущность-связь» (ER-модель). Диаграммы «сущность-связь» связаны со смысловым содержанием данных и независимо от физических параметров хранения данных [11, 16]. Также модели требуются для дальнейшего проектирования базы данных.

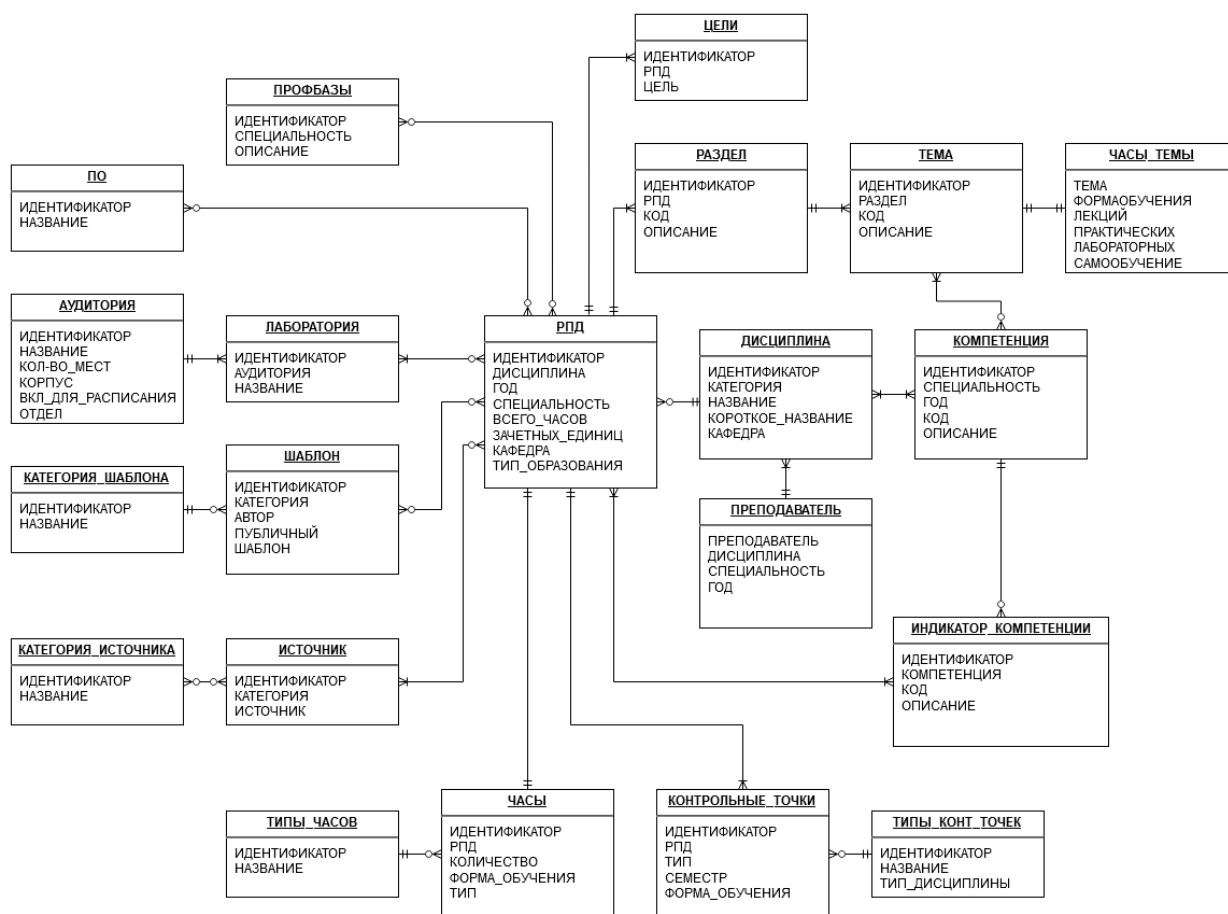


Рисунок 2.6 – Диаграмма «сущность-связь» (ER-модель)

На представленной ER-модели видно, что сущность «РПД» является основной для всей базы данных. Данная сущность необходима для хранения базовой информации о дисциплине, которая используется для формирования рабочей программы дисциплины. Одна дисциплина на конкретной специальности в конкретный год поступления на специальность должна иметь только один комплект учебно-методической документации. На диаграмме это показано как связь один ко многим между сущностями «Дисциплина» и

«РПД», а также атрибутами «Специальность» и «Год» у сущности «РПД». Такое представление связи выбрано потому, что сущность «Дисциплина» представляет собой справочник дисциплин, не привязанный к специальности или году.

К каждой дисциплине на определенной специальности в определенный год прикрепляется преподаватель (сущность «Преподаватель»), который будет заниматься разработкой учебно-методической документации для данной дисциплины. Разрабатывать документацию может только один преподаватель, но один преподаватель может заниматься разработкой документации для нескольких дисциплин.

Сущности «Компетенция» и «Индикатор_компетенции» созданы для представления справочника компетенций. Компетенции используются только в рамках конкретной специальности и конкретного года поступления. Каждая компетенция должна быть связана хотя бы с одной дисциплиной и соответственно с рабочей программой для данной дисциплины.

В рамках рабочей программы дисциплины, компетенции должны быть указаны для каждой темы из структуры дисциплины. Структура дисциплины представлена как сущность «Раздел» и «Тема». В рамках дисциплины может быть несколько разделов, каждый раздел содержит в себе дисциплины. Сущность «Компетенция» имеет связь с сущностью «Тема». Для всех тем также указывается количество часов, выделенное на данную тему (сущность «Часы_темы»).

Для всех рабочих программ должны быть указаны часы, выделенные на типы занятий – аудиторные и самостоятельные занятия, консультации, контрольные занятия. Для этого были созданы сущность «Типы_часов» для хранения справочника типа занятий и сущность «Часы» для хранения количества выделенных часов. Часы для одной дисциплины, но разных форм обучения (очно или заочно) различаются.

Сущности «Контрольные_точки» и «Типы_конт_точек» созданы для хранения списка контрольных точек дисциплины. В контрольные точки

входят зачеты, курсовые работы и экзамены. Контрольные точки отличаются для разных форм обучения.

Цели рабочей программы дисциплины должны храниться в сущности «Цели». У рабочей программы дисциплины может быть несколько целей.

Для хранения источников информации и профессиональных баз будут использоваться сущности «Источник», «Категория_источника» и «Профбазы». Все источники разделены на 3 категории – основные источники, дополнительные и ресурсы из сети Интернет. Каждый источник представляет из себя строку с названием источника, автором и др. информации об источнике. Один и тот же источник может быть использован в разных рабочих программах дисциплин. Профессиональные базы привязаны к конкретной специальности и также могут быть использованы в нескольких рабочих программах дисциплин.

Для справочника программного обеспечения создана сущность «ПО». В этой сущности будет храниться просто название программного обеспечения.

Справочник учебных лабораторий будет реализован в сущностях «Лаборатория» и «Аудитория». В одной аудитории может быть несколько лабораторий. Одна и та же лаборатория может использоваться для нескольких дисциплин. Для рабочей программы дисциплины также можно указать несколько лабораторий.

Для заполнения разделов с методическими рекомендациями и материально-техническим обеспечением в рабочих программах дисциплин, будут использоваться заготовленные шаблоны текста. Для хранения шаблонов созданы сущности «Категория_шаблона» и «Шаблон». У шаблона будет атрибут «Автор», чтобы ограничить права редактирования шаблонов. Также через атрибут «Публичный» можно указывать, является ли шаблон доступным для всех или нет. Сам текст шаблона будет храниться как текстовый объект.

Основываясь на созданной ER-модели, спроектируем СУБД-ориентированную модель данных (рисунок 2.7) [11, 16]. Части таблиц спроектированной базы данных расположены в разных схемах. Под схемой в

базе данных Oracle понимают учетную запись пользователя и объекты, принадлежащие этой учетной записи. Для взаимодействия с таблицами, находящимися в другой схеме, необходимо выдать соответствующие разрешения. В схеме «STUDY_PROCESS» расположены таблицы для справочника компетенций и таблица со списком аудиторий института. В схеме «RPD» находятся таблицы с базовой информацией о рабочей программе дисциплин, справочником источников информации, профессиональных баз и программного обеспечения, разделами и темами из структуры дисциплины, часами и контрольными точками дисциплины; таблицы, хранящие информацию о закреплении лабораторий, источников информации, программного обеспечения, компетенций и т.д. к рабочим программам дисциплин.

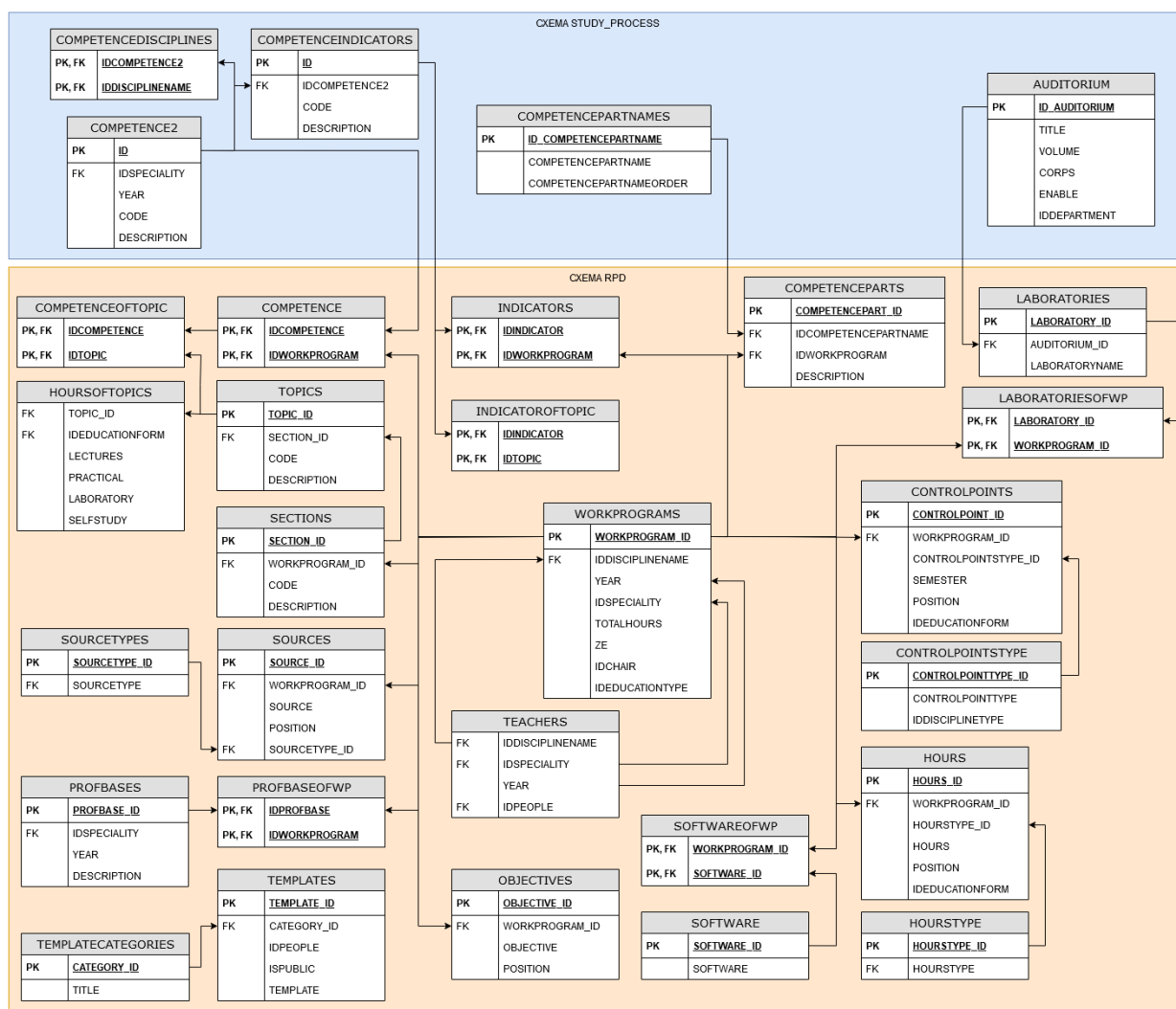


Рисунок 2.7 – Спроектированная СУБД-ориентированная модель данных

2.3 Разработка программного обеспечения

2.3.1 Структурная схема функций управления и обработки данных

Программный интерфейс web-приложения «Информационная система формирования учебно-методической документации для дисциплин» реализован на микросервисной архитектуре. При таком подходе приложение разбивается на несколько отдельных компонентов – микросервисов [20]. Каждый такой микросервис строится вокруг определенной бизнес-потребности и развертывается независимо от остальных компонентов.

На рисунке 2.8 представлено дерево реализованных сервисов и контроллеров разрабатываемого программного интерфейса web-приложения.

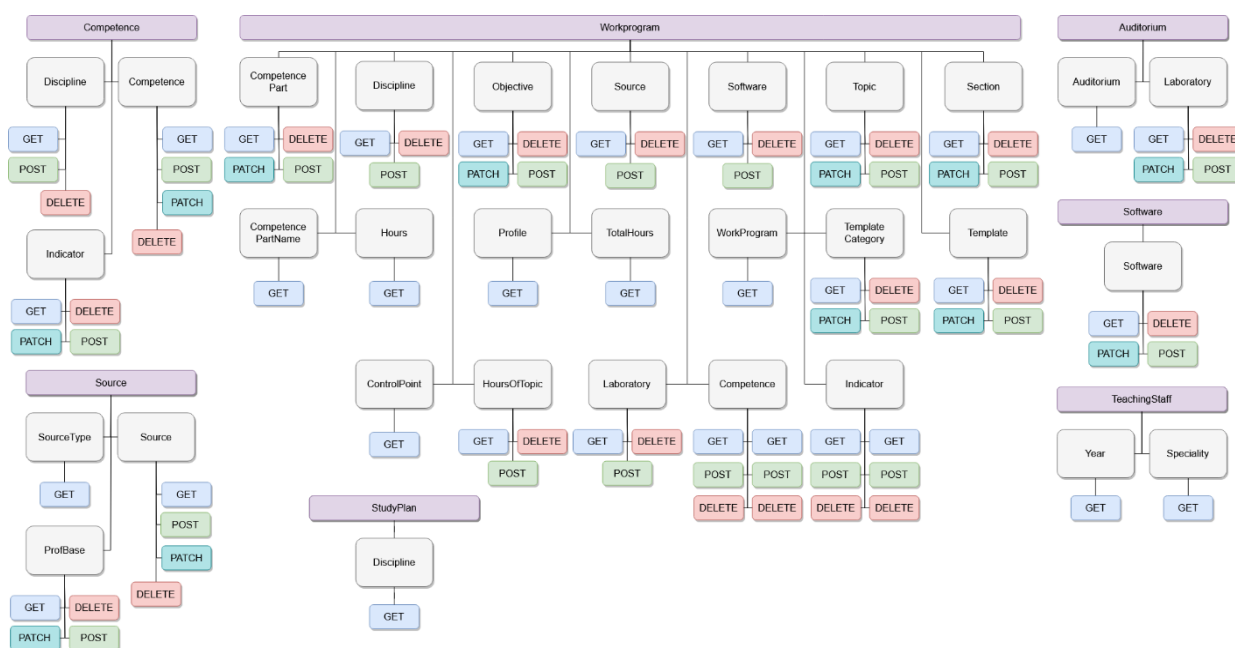


Рисунок 2.8 – Дерево реализованных сервисов и контроллеров разрабатываемого программного интерфейса web-приложения

Микросервис «Competence» предоставляет методы для работы со справочником компетенций. Через контроллер «Competence» можно получить список компетенций для конкретной специальности и года проведения набора

на специальность, создать новую компетенцию, изменить или удалить существующую. Контроллер «Discipline» для просмотра списка закрепленных за компетенцией дисциплин, закрепления и открепления дисциплин от компетенции. Контроллер «Indicator» позволяет получить список индикаторов достижения компетенции, добавить, изменить или удалить индикатор достижения компетенции.

В микросервисе «Auditorium» реализованы два контроллера: «Auditorium» с методом для получения списка аудиторий в институте и «Laboratory», для получения списка лабораторий, созданию новых, изменению и удалению существующих лабораторий.

Для работы со списком источников информации (например, литература) разработан микросервис «Source». В микросервисе есть контроллер «Source», позволяющий получать общий список источников, добавлять в него новые источники, изменять или удалять существующие. Контроллер «SourceType» позволяет получить список категорий, в которые могут быть помещены источники уже в рабочей программе дисциплины. Контроллер «ProfBase» позволяет работать с профессиональными базами. Для него также доступны методы для получения списка, создания, изменения и удаления профессиональных баз.

Для работы со списком программного обеспечения есть микросервис «Software» и контроллер «Software». В контроллере реализованы методы для получения списка программного обеспечения, добавления, изменения и удаления программного обеспечения.

Контроллеры «Year» и «Speciality» из микросервиса «TeachingStaff» позволяют получить список специальностей и года, в которые проводился набор на эти специальности.

Контроллер «Discipline» из микросервиса «StudyPlan» для получения списка дисциплин, преподаваемых на конкретной специальности.

Микросервис «Workprogram» содержит большое количество контроллеров, реализующих широкий функционал:

1. Контроллер «Workprogram», позволяет получить базовую информацию о дисциплине, необходимую для формирования рабочей программы дисциплины.
2. Контроллеры «Section» и «Topic» предоставляют методы для работы с разделами и темами для формирования структуры и содержания дисциплины (получение списка, создание, изменение и удаление разделов и тем).
3. К рабочей программе, а также разделам и темам можно прикрепить компетенции и индикаторы достижения компетенций. Эта возможность реализована в контроллерах «Competence» и «Indicator». Каждый контроллер дает возможность посмотреть список закрепленных компетенций и индикаторов, а также закрепить или открепить их.
4. Раздел «Знания, умения и навыки...» заполняется через методы, предоставляемые контроллерами «CompetencePart» и «CompetencePartName». Можно просмотреть список, добавленный для рабочей программы, создать, изменить или удалить запись.
5. Часы, выделяемые на дисциплину, а также распределение часов по темам разделов происходит через контроллеры «Hours», «TotalHours» и «HoursOfTopic».
6. Шаблоны и категории шаблонов добавляются и изменяются методами из контроллеров «Template», и «TemplateCategory».
7. Получить список профилей специальности можно в контроллере «Profile».
8. Контроллер «Software» предоставляет методы для получения списка добавленного в рабочую программу программного обеспечения, добавления и удаления программного обеспечения из рабочей программы.
9. В контроллере «Source» реализованы методы для получения списка добавленных в рабочую программу источников информации, добавления и удаления источников.

10. Также в рабочую программу можно добавить учебные лаборатории. Методы для этого реализованы в контроллере «Laboratory»: метод для получения списка добавленных в рабочую программу лабораторий и методы для добавления и удаления лабораторий из рабочей программы.

2.3.2 Описание программных модулей

Все разрабатываемые микросервисы, составляющие программный интерфейс web-приложения имеют одинаковую структуру. Рассмотрим структуру проекта на примере микросервиса «WorkProgram».

На рисунке 2.9 представлена структура проекта микросервиса «WorkProgram» в обозревателе решений Visual Studio.

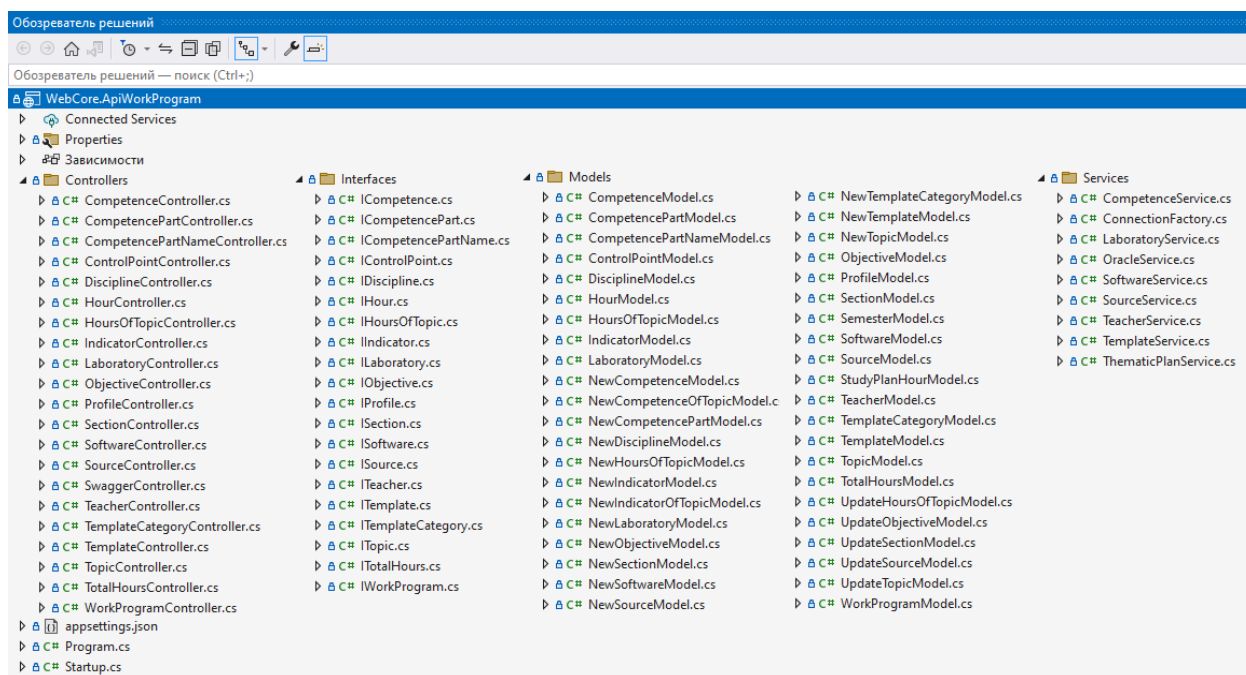


Рисунок 2.9 – Структура проекта микросервиса «WorkProgram» в обозревателе решений Visual Studio

Отдельные модули проекта распределены в различные пространства имен [17]. Это позволяет быстро ориентироваться в проекте, если необходимо найти какой-либо модуль или файл. Проект «WorkProgram» содержит несколько пространств имен:

1. Пространство `WorkProgram`. В пространство входят файлы `Program.cs` и `Startup.cs`. Все файлы расположены в корневой директории проекта. В этих файлах описываются параметры приложения для запуска (`ip` адрес, порт, параметры конфигурации `Swagger`), внедряются зависимости.

2. Пространство `Models`. Файлы, входящие в данное пространство имен размещены в директории `Models` внутри проекта. В этом пространстве имен находятся все модели, которые описывают структуру данных, получаемую в результате SQL запросов к базе данных, и возвращаемых клиентскому приложению в ответ на запрос от него. Модели представляют из себя обычные классы `C#` [3, 6].

3. Интерфейсы представляют ссылочный тип, определяющий набор методов, которые должны быть реализованы в некотором классе [5, 8]. Все интерфейсы расположены в директории `Interfaces` и пространстве имен `Interfaces`. Интерфейсы также необходимы для определения зависимостей внутри приложения.

4. Контроллеры программного интерфейса входят в пространство имен `Controllers` и расположены в директории с таким же названием. Контроллеры обрабатывают запросы, получаемые от клиентских приложений. В контроллерах обычно не описывается никакая бизнес-логика, а только логика обработки поступившего запроса [7, 20].

5. Сервисы в пространстве имен `Services` и директории `Services`. В файлах сервисов реализуются методы, определенные в интерфейсах. В сервисах описывается логика создания подключения к базе данных, формирования запроса к базе данных, обработки полученных результатов.

На рисунке 2.10 представлен пример кода модели, интерфейса и контроллера из микросервиса «`WorkProgram`».

`WorkProgramModel` – это модель данных рабочей программы дисциплин, содержащая базовую информацию о дисциплине. В модели `WorkProgramModel` описаны поля: идентификатор рабочей программы (`int IdWorkProgram`), идентификатор названия дисциплины (`int IdDisciplineName`),

год (int Year), идентификатор специальности (int IdSpeciality), общее количество часов, выделенное на дисциплину (int TotalHours), количество зачетных единиц (int ZE), идентификатор кафедры (int IdChair), название кафедры (string ChairName) и идентификатор типа образования (int IdEducationType).

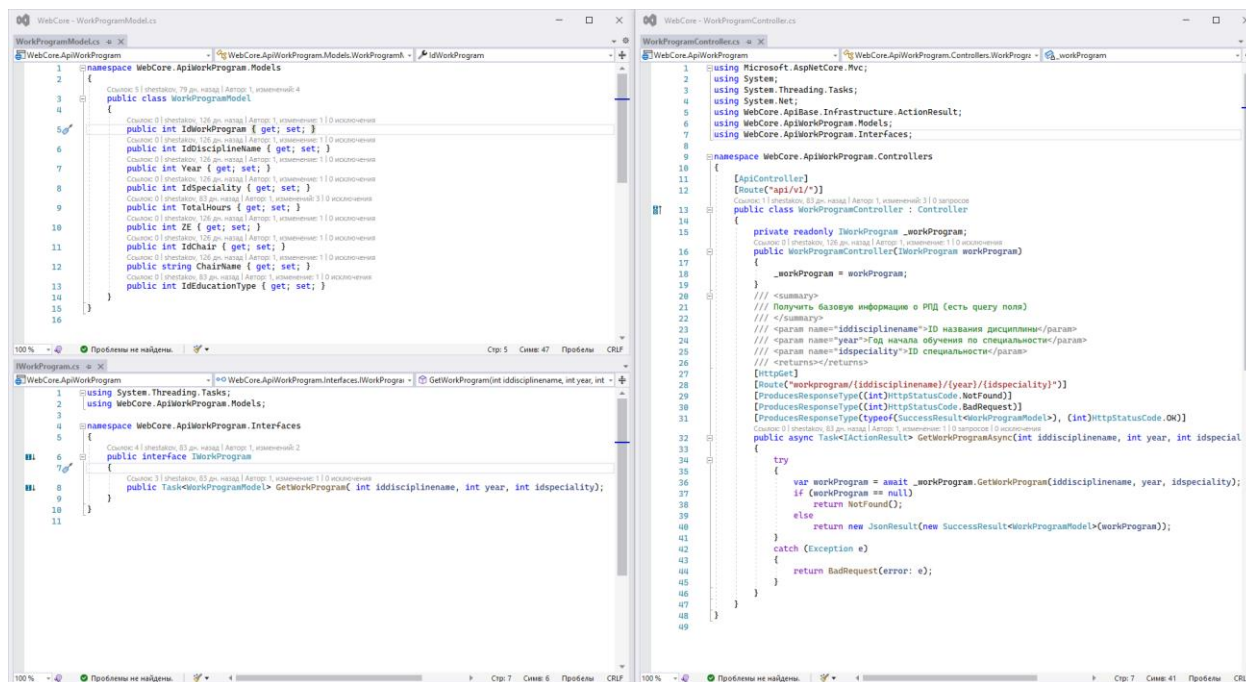


Рисунок 2.10 – Пример кода модели, интерфейса и контроллера из микросервиса «WorkProgram»

В контроллере WorkProgramController реализована логика только для HTTP-метода GET. При обращении к API по конечной точке «workprogram» вызывается метод обработки запроса. При запросе должны быть переданы параметры. На стороне клиентского приложения это выглядит как подстановка значений параметров в URL, например, «workprogram/121/2022/602», где «workprogram» это конечная точка контроллера, «121» это идентификатор названия дисциплины, 2022 – год, и 602 идентификатор специальности. Если параметры были получены и обработаны корректно, то контроллер обращается к сервису за данными и после возвращает их клиенту. В ином случае контроллер может вернуть клиенту код об ошибке.

Интерфейс `IWorkProgram` определяет методы, к которым будет обращаться контроллер API и которые должны быть реализованы в сервисе. В интерфейсе определен один метод `GetWorkProgram`. При вызове метода необходимо передать параметры: идентификатор названия дисциплины, год и идентификатор специальности.

На рисунке 2.11 представлен код метода получения базовой информации о дисциплине для формирования рабочей программы дисциплины в сервисе `WorkProgramService`.

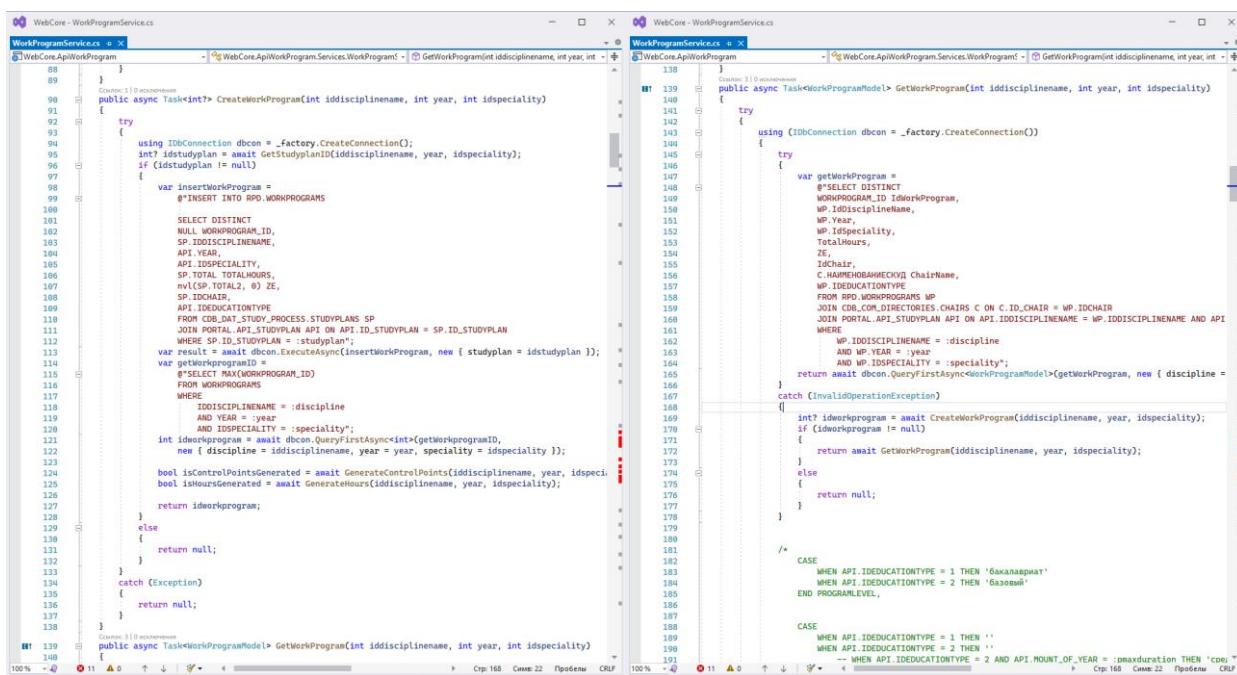


Рисунок 2.11 – Пример кода микросервиса «WorkProgram»

Контроллер микросервиса через интерфейс обращается к методу, реализованному в сервисе. В коде, представленном на рисунке 2.11, можно проследить процесс обращения к базе данных и обработке полученного результата. В реализуемом методе первым шагом производится попытка проверить, существует ли уже в базе данных информация, собранная и сгруппированная для указанной дисциплины и специальности. Если данные были получены, то они просто передаются контроллеру в виде объекта класса `WorkProgramModel`. В случае, если в базе данных не оказалось сгруппированной информации, вызывается метод, генерирующий данную

информацию. После успешного завершения процесса генерации, снова вызывается первый метод, возвращающий объект класса WorkProgramModel с полями, заполненными данными.

2.3.3 Схема взаимосвязи программных модулей и информационных файлов

Диаграмма развертывания моделирует физическое развертывание программных компонентов на аппаратных компонентах. Диаграмма развертывания программного интерфейса и web-приложения «Информационная система формирования учебно-методической документации для дисциплин» представлена на рисунке 2.12.

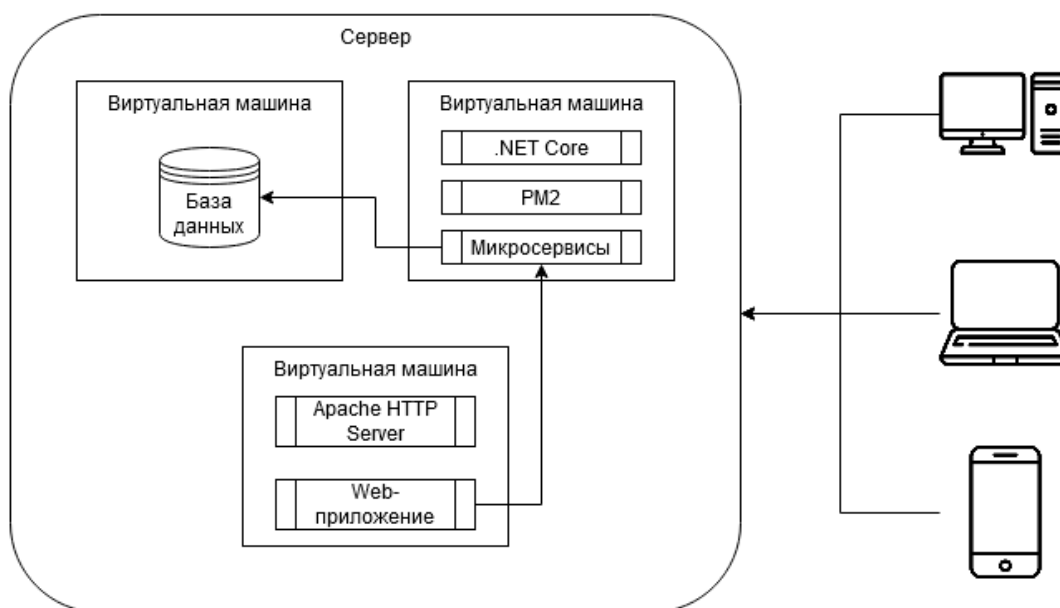


Рисунок 2.12 – Диаграмма развертывания программного интерфейса и web-приложения

По диаграмме развертывания можно увидеть, что на физическом сервере развернуто несколько виртуальных машин, на которых запущены процессы приложения. Одна из виртуальных машин выделена для базы данных, на второй установлена платформа .NET Core, менеджер процессов PM2 и микросервисы. Также на отдельной виртуальной машине развернут HTTP

сервер и web-приложение. Web-приложение обращается посредством HTTP-запросов к программному интерфейсу, программный интерфейс используя SQL общается с базой данных [18].

На диаграмме также видно, что доступ к web-приложению предоставляется с любого устройства (персональный компьютер, ноутбук, смартфон), подключенного к сети Интернет.

2.3.4 Компоненты пользовательского интерфейса

В разрабатываемом программном интерфейсе web-приложения «Информационная система формирования учебно-методической документации для дисциплин» не применяются графические интерфейсы, с которыми должен взаимодействовать непосредственно конечный пользователь, так как это исключительно серверное приложение.

Администратор приложения может получить доступ к консоли приложения и файлам логов приложения на сервере, в которые записываются все сообщения об ошибке в работе программного интерфейса.

Фреймворк Swagger автоматически генерирует документацию для приложения [1]. Доступ к документации предоставляется через графический web-интерфейс. Документация требуется для frontend-разработчика. Swagger предоставляет возможность ознакомиться со списком ответов, которые может вернуть сервер на запрос, в том числе с моделью данных, возвращаемых при успешном выполнении запроса. Также в Swagger можно протестировать работу программного интерфейса, выполнить HTTP-запрос и получить ответ в формате JSON. Данные в формате JSON не представляют никакого интереса для конечного пользователя web-приложения, но frontend-разработчик, на основе полученной модели данных, сможет разработать функционал, который обрабатывает данные и переносит их в графический интерфейс, с которым сможет взаимодействовать пользователь [18]. Пример графического

интерфейса, разрабатываемого frontend-разработчиков для web-приложения представлен на рисунке 2.13.

Компетенции 2022 Прикладная информатика

Список компетенций		Список индикаторов		Список дисциплин	
Код	Описание	Код	Описание	Дисциплина	
ОПК-1	способен применять естественнонаучные и общетеchnические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК 3.1.	Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	Алгоритмизация и программирование	
ОПК-2	способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК 3.2.	Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.	Вычислительные системы, сети и телекоммуникации	
ОПК-3	способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК 3.3.	Владет навыками подготовки обзоров, аннотаций и библиографии в области цифровых технологий и управления данными с учетом требований информационной безопасности.	Информационная безопасность	
ОПК-4	способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью			Информационные системы и технологии	
	способен устанавливать программное обеспечение			Теория вероятностей и математическая статистика	

Рисунок 2.13 – Пример интерфейса модуля web-приложения «Справочник компетенций» с загруженным списком компетенций

2.4 Компьютерно-сетевое обеспечение

В Рубцовском институте оборудовано несколько аудиторий со стационарными персональными компьютерами, две аудитории с мобильными персональными компьютерами (ноутбуками). На каждой кафедре и в других подразделениях института оборудованы рабочие места с персональными компьютерами. Характеристики установленных персональных компьютеров:

- 1) в учебных аудиториях:
 - процессор AMD Ryzen 2200G with Radeon Vega 8 Graphics (14 нм, 3.5 МГц – 3.7 МГц, 4 ядра, 4 потока, встроенное графическое ядро Radeon RX Vega 8 Graphics);
 - 8 ГБ оперативной памяти DDR4;
 - SSD накопитель 240 ГБ;

- 2 монитора AOC i2475Pхqu (24 дюйма, 60 Гц, IPS, 1920x1080);
- 2) на рабочих местах в подразделениях:
 - процессор Intel Pentium E5500 (45 нм, 2.8 ГГц, 2 ядра) или Intel Pentium П4520 (14 нм, 3.6 ГГц, 2 ядра, 2 потока);
 - 2 ГБ оперативной памяти DDR3 или 4 ГБ оперативной памяти DDR4;
 - HDD накопитель 80 ГБ или SSD накопитель 120 ГБ.

Все компьютеры в учебных аудиториях и на рабочих местах в подразделениях подключены к локальной сети института и имеют доступ к общим сетевым ресурсам в локальной сети института.

Локальная сеть института разделена на несколько подсетей:

- 1) подсеть для административных подразделений;
- 2) подсеть для учебных аудиторий;
- 3) подсеть для второго корпуса института;
- 4) подсеть для серверов и отдела технического и программного обеспечения.

Администрирование доступа к персональным компьютерам и сетевым ресурсам происходит с помощью Active Directory. Для этой задачи студентам и сотрудникам института выдаются персональные логины и генерируются пароли от учетной записи. Студенты и преподаватели могут самостоятельно изменить пароль от учетной записи на любом компьютере, подключенном к домену института или у системного администратора отдела технического и программного обеспечения.

Сеть первого и второго корпусов института соединена оптоволоконным кабелем. Внутри зданий институтов используется витая пара категории 6 и различное коммутационное оборудование (например, D-Link DGS-1024C). Для подключения к сети мобильных компьютеров в здании института установлены точки беспроводного доступа (например, D-Link DAP-2690). Доступ к беспроводным точкам доступа есть только у устройств, принадлежащих институту, ограничение доступа производится по белому

списку MAC адресов. Для студентов и сотрудников института также доступны две беспроводные точки доступа к сети Интернет в фойе института. Доступ к точкам предоставляется через логин и пароль от учетной записи в сети института (сервер FreeRADIUS).

2.5 Обеспечение информационной безопасности

Для обеспечения информационной безопасности в Рубцовском институте разработаны и применяются следующие организационные документы, описывающие положения и политики информационной безопасности института в различных процессах [14]:

1) положение об электронной информационно-образовательной среде Рубцовского института (филиала) АлтГУ, принято приказом № 227-1/п от 30 октября 2017 года;

2) положение о личном кабинете пользователя и корпоративной электронной почтовой системе Рубцовского института (филиала) АлтГУ, принято приказом № 228-1/п от 30 октября 2017 года;

3) концепция информатизации Рубцовского института (филиала) АлтГУ, утверждена решением Ученого совета Рубцовского института от 21 июня 2016 года;

4) положение об обработке персональных данных сотрудников Рубцовского института (филиала) АлтГУ, принято приказом № 324-1/п от 17 сентября 2019 года.

В соответствии с описанными положениями в институте реализуются следующие меры по информационной безопасности:

- серверная комната института оснащена металлической противопожарной дверью, закрываемой на ключ;
- ключ от серверной комнаты хранятся у начальника отдела технического и программного обеспечения института и выдается только

сотрудникам подразделения, и только в рамках проведения профилактических или восстановительных мероприятий для оборудования;

- у сотрудников других подразделений, в том числе администрации института, доступа в серверную комнату нет;

- предоставление доступа посторонним лицам в серверную комнату института производится исключительно по письменному разрешению начальника отдела технического и программного обеспечения и только в присутствии одного из сотрудников подразделения;

- локальная сеть института физически разделена на отдельные подсети (отдельные коммутаторы и порты маршрутизатора для каждой подсети);

- доступ к ресурсам между разными подсетями регулируется на программно-аппаратном уровне. Из подсети для учебных аудиторий нет доступа к ресурсам в подсети административных подразделений;

- в сети института используется домен Active Directory;

- доступ к персональным компьютерам на рабочих местах предоставляется по персональным логинам и паролям пользователей;

- в Active Directory установлены минимальные требования к паролям пользователей.

3 Оценка эффективности внедрения информационной системы

3.1 Общие положения

При разработки любой новой информационной системы для организации следует учитывать эффективность от ее внедрения.

Оценка эффективности информационных систем для каждого отдельного предприятия не разрабатывается, если нет в наличии оценок эффективности, которые представляют собой комплексную информационную систему. В то же время отдельные составляющие информационных систем тоже не рассматриваются без такой оценки.

Эффективность ИС – это свойство системы выполнять поставленную цель в заданных условиях использования и с определенным качеством. Эта характеристика отражает:

- действенность системы, т.е. степень соответствия ИС своему назначению (прагматическая эффективность);
- техническое совершенство ИС (техническая эффективность);
- простоту, технологичность разработки и создание системы (технологическая эффективность);
- удобство использования и обслуживания системы (эксплуатационная эффективность);
- улучшение и облегчение условий труда, изменение его содержания, развитие творческих функций, способностей и потребностей людей, преодоление существенных различий в труде и др. (социальная эффективность);
- экономическую целесообразность внедрения ИС, т.е. целесообразность произведенных на создание и функционирование системы затрат (экономическая эффективность).

Понятие эффективности связано с получением некоторого полезного результата – эффекта использования.

В соответствии с ГОСТ Р ИСО 9000-2001, эффективность функционирования ИС определяется соотношением результата (эффекта) и затраченными ресурсами. Приведенной оценкой затрат ресурсов выступает их стоимость. Затраты на функционирование ИС состоят из:

- стоимости приобретения программной платформы;
- стоимости доработки;
- стоимости внедрения;
- стоимости системного и вспомогательного программного обеспечения;
- стоимости аппаратного и сетевого обеспечения ИС;
- количества циклов (лет) эксплуатации;
- стоимости эксплуатации.

Основные задачи, стоящие при создании ИС – минимизация стоимости и обеспечение требуемого качества ИС.

Качество – это совокупность свойств системы, обуславливающих возможность ее использования для удовлетворения определенных потребностей пользователей в соответствии с ее назначением.

Основными показателями качества ИС являются:

- надежность;
- достоверность;
- безопасность.

Надежность – свойство системы сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных условиях применения.

Надежность ИС является средством обеспечения актуальной и достоверной информации на выходе системы.

Достоверность функционирования – свойство системы, обуславливающее безошибочность производимых ею преобразований

информации. Достоверность функционирования ИС полностью определяется и измеряется достоверностью ее результатной информации.

Безопасность – свойство, заключающееся в способности системы обеспечить конфиденциальность и целостность информации, т.е. защиту информации от несанкционированного доступа.

В любой сфере человеческой деятельности оценка эффективности внедрения любой новой техники, технологий и ИС осуществляется с помощью множества показателей. К ним относятся показатели прагматической, технической, эксплуатационной, социальной и экономической эффективности.

3.2 Показатели эффективности

Оценка эффективности внедрения любой новой технологии, информационных систем осуществляется с помощью множества показателей.

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.

С помощью показателей технической эффективности можно оценить техническое совершенство системы, оценить научно-технический уровень организации и функционирования 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.

Обобщающими показателями эффективности ИС являются показатели экономической эффективности. Обычно в качестве показателей экономической эффективности используются:

- 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.

Экономический эффект – это результат внедрения какого-либо мероприятия, выраженный в стоимостной форме, в виде экономии от его осуществления. Основными источниками экономии являются:

- улучшение показателей их основной деятельности, происходящее в результате использования программного изделия;
- повышение технического уровня, качества и объёмов вычислительных работ;
- увеличение объёмов и сокращение сроков переработки информации;
- повышение коэффициента использования вычислительных ресурсов, средств подготовки и передачи информации;
- уменьшение численности персонала, занятого обработкой исходных данных, переработкой и получением необходимой информации;
- снижение затрат на эксплуатационные материалы.

Срок окупаемости – показатель эффективности использования капиталовложений, представляет собой период времени, в течение которого произведенные затраты на программные изделия окупаются полученным эффектом.

Частные показатели экономической эффективности необходимы для оценки частного экономического эффекта, получаемого по отдельным источникам экономии.

Например: сокращение доли неквалифицированного и ручного труда на объекте экономики (ОЭ), сокращение материальных и энергетических затрат (повышение наукоемкости продукции), повышение производительности труда работников ОЭ, снижение уровня «бумажного» документооборота, повышение скорости производственных и экономических процессов на ОЭ и др.

Основные показатели для расчета экономической эффективности:

- показатель трудоемкости обработки информации;
- показатель эксплуатационных стоимостных затрат;
- экономический эффект;
- текущие затраты пользователя;
- экономия текущих затрат при автоматизации;
- относительная годовая экономия затрат на материалы.

3.3 Расчет экономической эффективности

Для оценки эффективности внедрения web-приложения «Информационная система формирования учебно-методической документации для дисциплин», частью которого является разработанный программный интерфейс, необходимо сравнить временные затраты преподавателей, затрачиваемые на формирование учебно-методической документации для дисциплин:

1) без применения web-приложения для формирования учебно-методической документации для дисциплин;

2) с применением web-приложения для формирования учебно-методической документации для дисциплин.

Формированием учебно-методической документации в Рубцовском институте занимаются преподаватели института.

В таблице 3.1 представлены приблизительные расчеты временных затрат преподавателей на формирование учебно-методической документации для дисциплины.

Таблица 3.1 – Расчеты временных затрат преподавателей на формирование учебно-методической документации для дисциплины.

Работник	Название работ	Базовая ИС T_0		Предлагаемая ИС, T_i	
		В день, минут	В год, часов	В день, минут	В год, часов
Преподаватель	Получение и анализ данных из учебного плана	120	2	0	0
	Получение списка материально-технического обеспечения	30	0,5	15	0,25
	Получение списка программного обеспечения	30	0,5	15	0,25
	Выбор источников информации	120	2	30	0,5
	Выбор компетенций для дисциплины	60	1	0	0
	Формирование структуры и содержания дисциплины	60	1	30	0,5
	Формирование файла Microsoft Word с РПД	60	1	0	0
	Формирование списка заданий для ФОС	90	1,5	30	0,5

Продолжение таблицы 3.1

	Формирование файла Microsoft Word с ФОС	30	0,5	0	0
	Формирование аннотации к РПД	30	0,5	0	0
Итого преподаватель			10,5		2
Итого по всем			10,5		2

Показатель T_0 означает показатель величины трудоемкости без применения web-приложения для формирования учебно-методической документации для дисциплин, T_j – показатель величины трудоемкости с применением web-приложения для формирования учебно-методической документации для дисциплин.

Для нахождения показателя снижения трудовых затрат ΔT за год, воспользуемся формулой (3.1).

$$\Delta T = T_0 - T_j \quad (3.1)$$

В результате расчетов получим значение ΔT равное 8,5 часов.

По формуле (3.2) рассчитаем коэффициент относительного снижения трудовых затрат, который покажет, на какую долю снизятся трудовые затраты.

$$K_t = \Delta T / T_0 \quad (3.2)$$

В результате вычислений получим коэффициент 0,8095 или если перевести в процентное соотношение, то 80,95%.

Снижение трудовых затрат происходит по нескольким причинам:

1. Теперь основная часть поиска и анализа базовой информации о дисциплине из учебного плана, необходимой для формирования учебно-методической документации, происходит на сервере в автоматическом

режиме. Преподаватель не тратит время на поиск и перепроверку данной информации из таблиц Microsoft Excel.

2. С внедрением web-приложения, для внесения компетенций в рабочую программу дисциплины необходимо только обратиться к справочнику компетенций, в которых будут показаны закрепленные методистом компетенции для данной дисциплины.

3. Шаблоны для заполнения разделом с материально-техническим обеспечением и методическими рекомендациями сокращают время на составление данных разделов. При исправлении ошибки в тексте шаблона текст поменяется во всех документах автоматически.

4. Теперь для поиска и закрепления источников информации можно использовать справочник источников. Достаточно один раз внести источник в справочник, после чего закреплять его к разным рабочим программам дисциплин.

5. Приложение генерирует PDF файлы с рабочей программой дисциплины, фондом оценочных средств и аннотацией по запросу пользователя.

Так как формирования учебно-методической документации для дисциплины является обязанностью преподавателей, то данный процесс не имеет отдельной доплаты. Внедренное web-приложение не влияет на количество расходуемых материалов (бумага, картриджи для принтеров, канцелярские товары) в процессе формирования учебно-методической документации. Новые расходы, связанные с внедрением и эксплуатацией web-приложения, не появятся. В связи с этим проводить оценку эффективности от внедрения приложения с точки зрения финансовых затрат не имеет смысла.

Рассчитаем затраты на разработку программного интерфейса и web-приложения. В расчетах будем учитывать затраты на разработку проекта приложения, затраты на написание и отладку программного кода, затраты на разработку документации и внедрение. Для приблизительных расчетов затрат на разработку был взят часовой оклад в 125 рублей в час. При расчетах не

учитывались отчисления в страховые фонды и выплаты районного коэффициента. Также при расчетах не учитывались расходы на амортизацию оборудования и электроэнергию. Приблизительные расчеты затрат на разработку приложения представлены в таблице 3.2

Таблица 3.2 Расчеты затрат на разработку приложения

Статья	Дней	Часов	Стоимость работы за час, рублей	Итого, рублей
Разработка проекта приложения	5	20,00	125,00	2500,00
Написание и отладка программного кода программного интерфейса	80	360,00	125,00	45000,00
Написание и отладка программного кода web- приложения	80	360,00	125,00	45000,00
Разработка документации	5	20,00	125,00	2500,00
Внедрение приложения	5	10,00	125,00	1250,00
Итого				96250,00

Приблизительные затраты на разработку программного интерфейса и web-приложения составили 96250,00 рублей.

ЗАКЛЮЧЕНИЕ

В рамках данной выпускной квалификационной работы была поставлена цель разработать программный интерфейс web-приложения «Информационная система формирования учебно-методической документации для дисциплин» (на примере Рубцовского института (филиала) АлтГУ).

Для достижения цели выпускной квалификационной работы были поставлены и выполнены следующие задачи:

- 1) рассмотрены и проанализированы существующее состояние предметной области, характеристики объекта и системы управления, характеристики используемых информационных систем;
- 2) выявлены недостатки функционирования объекта и предложены предложения по устранению выявленных недостатков;
- 3) составлено описание состава и структуры разрабатываемой системы, связей между функциональными подсистемами;
- 4) составлено описание элементов и модулей информационного, технического, программного и других обеспечивающих подсистем;
- 5) разработан программный интерфейс web-приложения;
- 6) составлена оценка эффективности внедрения информационной системы.

Так как все поставленные задачи были выполнены, можно сделать вывод, что цель данной выпускной квалификационной работы была достигнута. Разработанный программный интерфейс в дальнейшем будет использоваться для разработки web-приложения «Информационная система формирования учебно-методической документации для дисциплин» и внедрения этого приложения в процесс формирования учебно-методической документации в Рубцовском институте (филиале) АлтГУ. Планируется, что с внедрением web-приложения «Информационная система формирования

учебно-методической документации для дисциплин» повысится эффективность разработки учебно-методической документации:

- за счет использования справочников, хранящихся в базе данных, время, затрачиваемое на разработку учебно-методической документации, существенно сократится;

- web-приложение будет автоматически загружать и группировать информацию из базы данных информационной системы «Учебные планы», благодаря чему преподаватели будут допускать меньше ошибок при работе с учебными планами;

- приложение будет автоматически генерировать PDF файлы с рабочими программами дисциплин, фондами оценочных средств и аннотациями к рабочим программам по запросу пользователя. Преподавателям больше не придется вручную формировать документы в файлах Microsoft Word.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. API Documentation & Design Tools for Teams | Swagger : официальный сайт. – Текст : электронный. – URL: <https://swagger.io>. – Загл. с экрана.
2. Dapper Tutorial : официальный сайт. – Текст : электронный. – URL: <https://dapper-tutorial.net>. – Загл. с экрана.
3. Price, Mark. C# 10 and .NET 6 – Modern Cross-Platform Development. Sixth Edition. / Mark Price. – Birmingham : Packt, 2021. – 825 с. – Текст : электронный. – URL: <https://subscription.packtpub.com/book/mobile/9781801077361>. – Загл. с экрана.
4. Visual Studio IDE и редактор кода для разработчиков и групп, работающих с программным обеспечением : официальный сайт. – Текст : электронный. – URL: <https://visualstudio.microsoft.com>. – Загл. с экрана.
5. Албахари, Джозеф. C# 9.0. Карманный справочник / Джозеф Албахари, Бен Албахари. – Санкт-Петербург : Диалектика, 2021. – 256 с. – Текст : электронный. – URL: <https://www.labirint.ru/books/801590>. – Загл. с экрана.
6. Арораа, Г. Паттерны проектирования для C# и платформы .NET Core / Г. Арораа, Д. Чилберто. – Санкт-Петербург : Издательство Питер, 2021. – 352 с. – Текст : электронный. – URL: <https://www.piter.com/collection/all/product/patterny-proektirovaniya-dlya-c-i-platformy-net-core>. – Загл. с экрана.
7. Гриффитс, И. Програмируем на C# 8.0. Разработка приложений / И. Гриффитс. – Санкт-Петербург : Издательство Питер, 2021. – 944 с. – Текст : электронный. – URL: <https://www.piter.com/collection/all/product/izuchaem-c-80-razrabotka-prilozheniy>. – Загл. с экрана.
8. Клири, С. Конкурентность в C#. Асинхронное, параллельное и многопоточное программирование / С. Клири. – Санкт-Петербург :

Издательство Питер, 2020. – 304 с. – Текст : электронный. – URL: <https://www.piter.com/collection/all/product/konkurentnost-v-c-asinhronnoe-parallelnoe-i-mnogopotochnoe-programmirovaniye-2-e-mezhd-izd>. – Загл. с экрана.

9. Лаборатория ММИС : официальный сайт. – Шахты. – Текст : электронный. – URL: <https://www.mmis.ru>. – Загл. с экрана.

10. Майер, С. Ф. Анализ систем автоматизации формирования рабочих программ учебных дисциплин. / С. Ф. Майер, Г. В. Муратова. – Текст: электронный // КиберЛенинка : [Электронный ресурс]. – 2021. – URL: <https://cyberleninka.ru/article/n/analiz-sistem-avtomatizatsii-formirovaniya-rabochih-programm-uchebnyh-distiplin>. – Загл. с экрана.

11. Нестеров, С. А. Базы данных : учебник и практикум для вузов / С. А. Нестеров. – Москва : Издательство Юрайт, 2022. – 230 с. – (Высшее образование). – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/489693>. – Загл. с экрана.

12. О персональных данных: Закон РФ от 27.07.2006 №152-ФЗ (ред. от 02.07.2021). – Текст : электронный // КонсультантПлюс. – Режим доступа: по подписке.

13. Прайс, М. С# 10 и .NET 5. Разработка и оптимизация / М. Прайс. – Санкт-Петербург : Издательство Питер, 2022. – 832 с. – Текст : электронный. – URL: <https://www.piter.com/collection/all/product/c-9-i-net-5-razrabotka-i-optimizatsiya>. – Загл. с экрана.

14. Рубцовский институт филиал АлтГУ : официальный сайт. – Рубцовск. – Текст : электронный. – URL: <https://rb.asu.ru/>. – Загл. с экрана.

15. Средства разработчика, техническая документация и примеры кода | Microsoft Docs : официальный сайт. – Текст : электронный. – URL: <https://docs.microsoft.com>. – Загл. с экрана.

16. Стружкин, Н. П. Базы данных: проектирование : учебник для вузов / Н. П. Стружкин, В. В. Годин. – Москва : Издательство Юрайт, 2022. – 477 с. – (Высшее образование). – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/489099>. – Загл. с экрана.

17. Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для вузов / А. Ф. Тузовский. – Москва : Издательство Юрайт, 2022. – 206 с. – (Высшее образование). – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/490369>. – Загл. с экрана.
18. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для вузов / А. Ф. Тузовский. – Москва : Издательство Юрайт, 2022. – 218 с. – (Высшее образование). – Текст : электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/bcode/490128>. – Загл. с экрана.
19. ФГОС : официальный сайт. – Москва. – Текст : электронный. – URL: <https://fgos.ru/>. – Загл. с экрана.
20. Хорсдал, К. Микросервисы на платформе .NET / К. Хорсдал. . – Санкт-Петербург : Издательство Питер, 2018. – 352 с. – Текст : электронный. – URL: <https://www.piter.com/collection/all/product/mikroservisy-na-platforme-net>. – Загл. с экрана.