

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Иркутский государственный университет путей сообщения»  
(ФГБОУ ВО ИРГУПС)

УТВЕРЖДЕНА  
приказом и.о. ректора  
от «17» июня 2022 г. № 77

**Б1.О.06 Методология программной инженерии**

**рабочая программа дисциплины**

Специальность/направление подготовки – 09.04.04 Программная инженерия

Специализация/профиль – Методология разработки программно-информационных систем

Квалификация выпускника – Магистр

Форма и срок обучения – очная форма 2 года

Кафедра-разработчик программы – Информационные системы и защита информации

Общая трудоемкость в з.е. – 6

Часов по учебному плану (УП) – 216

Формы промежуточной аттестации

очная форма обучения:

зачет 2 семестр, экзамен 3 семестр, курсовая работа 3 семестр

**Очная форма обучения**

**Распределение часов дисциплины по семестрам**

Семестр	2	3	Итого
Вид занятий	Часов по УП	Часов по УП	Часов по УП
<b>Аудиторная контактная работа по видам учебных занятий/ в т.ч. в форме ПП*</b>	34	34	<b>68</b>
– лекции	17	17	<b>34</b>
– практические (семинарские)			
– лабораторные	17	17	<b>34</b>
<b>Самостоятельная работа</b>	38	74	<b>112</b>
<b>Экзамен</b>		36	<b>36</b>
<b>Итого</b>	<b>72</b>	<b>144</b>	<b>216</b>

ИРКУТСК

Электронный документ выгружен из ЕИС ФГБОУ ВО ИРГУПС и соответствует оригиналу

Подписант ФГБОУ ВО ИРГУПС Трофимов Ю.А.

00a73c5b7b623a969ccad43a81ab346d50 с 08.12.2022 14:32 по 02.03.2024 14:32 GMT+03:00

Подпись соответствует файлу документа



Рабочая программа дисциплины разработана в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по направлению подготовки 09.04.04 Программная инженерия, утвержденным Приказом Министерства науки и высшего образования Российской Федерации от 19.09.2017 № 932.

Программу составил(и):

К.э.н., доцент, заведующий кафедрой, Т. К. Кириллова

К.т.н., доцент, О. В. Курганская

Рабочая программа рассмотрена и одобрена для использования в учебном процессе на заседании кафедры «Информационные системы и защита информации», протокол от «17» июня 2022 г. № 12

Зав. кафедрой, к.э.н., доцент

Т.К. Кириллова

<b>1 ЦЕЛЬ И ЗАДАЧИ ДИСЦИПЛИНЫ</b>	
<b>1.1 Цель дисциплины</b>	
1	изучение методологии программной инженерии (ПИ)
<b>1.2 Задачи дисциплины</b>	
1	ознакомление обучаемых с состоянием и тенденциями развития программной инженерии как одного из прикладных направлений информационных технологий;
2	изучение методов ПИ

<b>2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП</b>	
Блок/часть ОПОП	Блок 1. Дисциплины / Обязательная часть
<b>2.1 Дисциплины и практики, на которых основывается изучение данной дисциплины</b>	
1	Б1.О.01 Логика и методология науки
2	Б1.О.02 Основы научных исследований
3	Б1.О.07 Конструирование компиляторов
4	Б1.О.10 Проектирование операционных систем
5	Б2.О.01(У) Учебная - ознакомительная практика
6	ФТД.01 Логика
<b>2.2 Дисциплины и практики, для которых изучение данной дисциплины необходимо как предшествующее</b>	
1	Б1.О.11 Моделирование
2	Б1.О.14 Системы искусственного интеллекта
3	Б2.О.01(У) Учебная - ознакомительная практика
4	Б2.О.02(Н) Производственная - научно-исследовательская работа в семестре
5	Б2.О.04(Пд) Производственная - преддипломная практика
6	Б3.01(Д) Выполнение выпускной квалификационной работы
7	Б3.02(Д) Защита выпускной квалификационной работы
8	ФТД.02 Принципы инженерного творчества

<b>3 ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ТРЕБОВАНИЯМИ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ</b>		
<b>Код и наименование компетенции</b>	<b>Код и наименование индикатора достижения компетенции</b>	<b>Планируемые результаты обучения</b>
ОПК-1 Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	ОПК-1.1 Знает математические, естественнонаучные и социально-экономические методы для использования в профессиональной деятельности	Знать: – формальные методы разработки программных систем (ПС)
		Уметь: – объяснить особенности методов разработки ПС
		Владеть: – представлением о формальных методах разработки ПО
	ОПК-1.2 Умеет решать нестандартные профессиональные задачи, в том числе в новой или незнакомой среде и в междисциплинарном контексте, с применением математических, естественнонаучных, социально-экономических и профессиональных знаний	Знать: – особенности методологии ПИ при решении прикладных задач разработки и сопровождения ПС
		Уметь: – решать отдельные задачи проектирования ПС
		Владеть: – навыками решения проектных задач ПИ
ОПК-1.3 Имеет навыки теоретического и экспериментального исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Знать: – теоретические и прикладные аспекты методологии ПИ	
	Уметь: – применять методы ПИ при решении прикладных задач	
	Владеть: – навыками применения методологии ПИ при разработке прикладного ПО	
ОПК-3 Способен анализировать	ОПК-3.1 Знает принципы, методы и средства анализа и	Знать: – как оформить самостоятельно изученный материал

профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями	структурирования профессиональной информации	Уметь: – представить материал преподавателю
		Владеть: – объяснить найденный материал
	ОПК-3.2 Умеет анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров	Знать: – ключевые вопросы дисциплины
		Уметь: – разъяснить ключевые вопросы преподавателю
	ОПК-3.3 Имеет навыки подготовки научных докладов, публикаций и аналитических обзоров с обоснованными выводами и рекомендациями	Владеть: – навыками письменного и устного изложения материала по дисциплине
		Знать: – проблематику ПИ
Уметь: – донести проблематику через отчеты, рефераты, эссе		
УК-6 Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки	УК-6.1 Знает основные принципы самовоспитания и самообразования, профессионального и личностного развития, исходя из этапов карьерного роста и требований рынка труда	Знать: – требования рынка труда к разработчикам программного обеспечения (ПО)
		Уметь: – приобретать новые знания в профессиональной области
		Владеть: – опытом личностного развития в своей профессиональной области за счет приобретения новых знаний, умений и навыков
	УК-6.2 Умеет планировать свое рабочее время и время для саморазвития, формулировать цели личностного и профессионального развития и условия их достижения, исходя из тенденций развития области профессиональной деятельности, индивидуально-личностных особенностей	Знать: – тенденции развития области своей профессиональной деятельности
		Уметь: – планировать свое рабочее время
	УК-6.3 Имеет практический опыт получения дополнительного образования, изучения дополнительных образовательных программ	Владеть: – навыками самоорганизации при изучении разделов дисциплины, предназначенных для самостоятельного изучения
		Знать: – интернет-источники материалов по дисциплине
		Уметь: – найти необходимый материал, представить его в отчете по лабораторной работе, реферате, эссе
		Владеть: – навыками самостоятельного поиска информации по дисциплине, ее обработке и представлении

#### 4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Очная форма				*Код индикатора достижения компетенции
		Семестр	Часы			
			Лек	Пр	Лаб	
<b>1.0</b>	<b>Раздел 1 Введение в дисциплину.</b>					
1.1	Тема 1. Понятие и задачи программной инженерии	2	2	3	6	ОПК-1.1 ОПК-1.3 ОПК-3.2 УК-6.1 УК-6.2 УК-6.3
1.2	Тема 2. Проектирование программного обеспечения	2	2	4	6	ОПК-1.2 ОПК-3.2
1.3	Тема 3. Жизненный цикл ПО от технического задания на разработку до завершения эксплуатации	2	2		6	ОПК-1.2 ОПК-3.1 ОПК-3.2 ОПК-3.3 УК-6.2

#### 4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Очная форма				*Код индикатора достижения компетенции
		Семестр	Часы			
			Лек	Пр	Лаб	
<b>2.0</b>	<b>Раздел 2 Стандарты разработки, технико-экономическое обоснование и требования к ПО. Планирование жизненного цикла. Проектирование ПО.</b>					
2.1	Тема 4. Стандарты и профили стандартов	2	2		6	ОПК-3.1 ОПК-3.2
2.2	Тема 5. Системное проектирование приложений	2	2	4	6	ОПК-1.1 ОПК-1.2 УК-6.2
2.3	Тема 6. Обоснование разработки ПО	2	2		6	ОПК-1.1 ОПК-3.2
2.4	Тема 7. Модели жизненного цикла и их особенности	2	2	4	6	ОПК-3.1
2.5	Тема 8. Применение визуального языка моделирования	2	3	3	6	ОПК-1.1 ОПК-1.3
	Форма промежуточной аттестации – зачет	2				
<b>3.0</b>	<b>Раздел 3 Проблемы разработки. Тестирование и документирование ПО.</b>					
3.1	Тема 9. Ресурсы ПО и управление ими	3	4	4	6	ОПК-1.2 ОПК-3.2 УК-6.2 УК-6.3
3.2	Тема 10. Риски разработки ПО. Отладка ПО	3	4	4	8	ОПК-1.2 ОПК-3.2
3.3	Тема 11. Верификация программных кодов	3	3	4	8	ОПК-1.1 ОПК-1.2
3.4	Тема 12. Сопровождение и мониторинг ПО	3	4		8	ОПК-3.2
3.5	Тема 13. Документирование ПО	3	2	4	9	ОПК-3.2 ОПК-3.3
	Форма промежуточной аттестации – экзамен	3		36		ОПК-1.1 ОПК-1.2 ОПК-1.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 УК-6.1 УК-6.2 УК-6.3
	Курсовая работа	3			25	ОПК-1.1 ОПК-1.2 ОПК-3.1 ОПК-3.3 УК-6.2 УК-6.3
	Итого часов (без учёта часов на промежуточную аттестацию)		34	34	112	

#### 5 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине оформлен в виде приложения № 1 к рабочей программе дисциплины и размещен в электронной информационно-образовательной среде Университета, доступной обучающемуся через его личный кабинет

#### 6 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

**6.1 Учебная литература**  
**6.1.1 Основная литература**

	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.1.1	Антамошкин, О. А. Программная инженерия. Теория и практика : учебник / О. А. Антамошкин. Красноярск : Сибирский федеральный университет (СФУ), 2012. - 247с. - Текст: электронный. - URL: <a href="https://biblioclub.ru/index.php?page=book&amp;id=363975">https://biblioclub.ru/index.php?page=book&amp;id=363975</a> (дата обращения: 14.09.2022)	Онлайн
6.1.1.2	Доррер, Г. А. Методология программной инженерии : учебное пособие / Г. А. Доррер. Красноярск : СибГУ им. академика М. Ф. Решетнёва, 2021. - 190с. - Текст: электронный. - URL: <a href="https://e.lanbook.com/book/195097">https://e.lanbook.com/book/195097</a> (дата обращения: 19.04.2023)	Онлайн
6.1.1.3	Крючкова, Е. Н. Программирование параллельных процессов : учебное пособие для студентов направления 09.04.04 «программная инженерия» (магистратура) / Е. Н. Крючкова, С. М. Старолетов. Барнаул : АлтГТУ, 2020. - 206с. - Текст: электронный. - URL: <a href="https://e.lanbook.com/book/292796">https://e.lanbook.com/book/292796</a> (дата обращения: 19.04.2023)	Онлайн
6.1.1.4	Птицына, Л. К. Методология программной инженерии : учебное пособие / Л. К. Птицына, И. Б. Бондаренко. Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2021. - 55с. - Текст: электронный. - URL: <a href="https://e.lanbook.com/book/279605">https://e.lanbook.com/book/279605</a> (дата обращения: 19.04.2023)	Онлайн
<b>6.1.2 Дополнительная литература</b>		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.2.1	Алексеева, Н. В. Управление проектами: учебно-методическое пособие / Н. В. Алексеева. Москва : РТУ МИРЭА, 2021. - 74с. - Текст: электронный. - URL: <a href="https://e.lanbook.com/book/171533">https://e.lanbook.com/book/171533</a> (дата обращения: 19.04.2023)	Онлайн
<b>6.1.3 Учебно-методические разработки (в т. ч. для самостоятельной работы обучающихся)</b>		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.3.1	Кириллова, Т.К. Методические указания по изучению дисциплины Б1.О.06 Методология программной инженерии по направлению подготовки 09.04.04 Программная инженерия, профиль Методология разработки программно-информационных систем / Т.К. Кириллова; ИрГУПС. – Иркутск : ИрГУПС, 2023. – 17 с - Текст: электронный. - URL: <a href="https://www.irgups.ru/eis/for_site/umkd_files/mu_8042_1406_2022_1_signed.pdf">https://www.irgups.ru/eis/for_site/umkd_files/mu_8042_1406_2022_1_signed.pdf</a>	Онлайн
<b>6.2 Ресурсы информационно-телекоммуникационной сети «Интернет»</b>		
6.2.1	Научная электронная библиотека eLIBRARY.RU — <a href="https://elibrary.ru/">https://elibrary.ru/</a>	
6.2.2	Электронная библиотека Учебно-методического центра по образованию на железнодорожном транспорте «ЭБ УМЦ ЖДТ» — <a href="https://umczdt.ru/books/">https://umczdt.ru/books/</a>	
6.2.3	Электронно-библиотечная система «Издательство Лань», <a href="https://e.lanbook.com/">https://e.lanbook.com/</a>	
6.2.4	Электронно-библиотечная система «Образовательная платформа ЮРАЙТ», <a href="https://urait.ru/">https://urait.ru/</a>	
6.2.5	Электронно-библиотечная система «Университетская библиотека онлайн», <a href="https://biblioclub.ru/">https://biblioclub.ru/</a>	
<b>6.3 Программное обеспечение и информационные справочные системы</b>		
<b>6.3.1 Базовое программное обеспечение</b>		
6.3.1.1	Microsoft Windows Professional 10, государственный контракт от 20.07.2021 № 0334100010021000013-01	
6.3.1.2	Microsoft Office Russian 2010, государственный контракт от 20.07.2021 № 0334100010021000013-01	
6.3.1.3	FoxitReader, свободно распространяемое программное обеспечение <a href="http://free-software.com.ua/pdf-viewer/foxit-reader/">http://free-software.com.ua/pdf-viewer/foxit-reader/</a>	
6.3.1.4	Adobe Acrobat Reader DC свободно распространяемое программное обеспечение <a href="https://get.adobe.com/ru/reader/enterprise/">https://get.adobe.com/ru/reader/enterprise/</a>	
6.3.1.5	Яндекс. Браузер. Прикладное программное обеспечение общего назначения, Офисные приложения, лицензия – свободно распространяемое программное обеспечение по лицензии BSD License	
<b>6.3.2 Специализированное программное обеспечение</b>		
6.3.2.1	Python 3.9, свободно распространяемое программное обеспечение <a href="https://docs.python.org/3/license.html">https://docs.python.org/3/license.html</a>	
6.3.2.2	Dev-C++, свободная интегрированная среда разработки приложений для языков программирования C/C++, <a href="https://code-live.ru/post/dev-cpp-free-cpp-ide-for-windows/">https://code-live.ru/post/dev-cpp-free-cpp-ide-for-windows/</a> Microsoft MS Project, государственный контракт от 20.07.2021 № 0334100010021000013-01 Visual Studio 2022	

	Community, образовательная лицензия, <a href="https://visualstudio.microsoft.com/ru/license-terms/vs2022-ga-community/">https://visualstudio.microsoft.com/ru/license-terms/vs2022-ga-community/</a> Visual Studio Code, образовательная лицензия, <a href="https://code.visualstudio.com/license">https://code.visualstudio.com/license</a> NetBeans IDE, свободная лицензия Apache License 2.0 <a href="https://www.apache.org/licenses/">https://www.apache.org/licenses/</a> Java Virtual Machine, свободная лицензия Oracle Java SE <a href="https://www.oracle.com/downloads/licenses/javase-license1.html">https://www.oracle.com/downloads/licenses/javase-license1.html</a>
<b>6.3.3 Информационные справочные системы</b>	
6.3.3.1	Не предусмотрены
<b>6.4 Правовые и нормативные документы</b>	
6.4.1	Не предусмотрены

<b>7 ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ УЧЕБНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ</b>	
1	Корпуса А, Б, В, Г, Д, Е ИрГУПС находятся по адресу г. Иркутск, ул. Чернышевского, д. 15; корпус Л ИрГУПС находится – по адресу г. Иркутск, ул. Лермонтова, д.80
2	Учебная аудитория Д-518* для проведения лекционных и практических занятий, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, (ноутбук переносной). Для проведения занятий имеются учебно-наглядные пособия (презентации).
3	Класс А-401 "Деловых игр" для проведения практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, компьютеры с подключением к сети Интернет, обеспечивающие доступ в электронную информационно-образовательную среду ИрГУПС. Для проведения занятий имеются учебно-наглядные пособия (презентации).
4	Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к информационно-телекоммуникационной сети «Интернет», и обеспечены доступом в электронную информационно-образовательную среду ИрГУПС. Помещения для самостоятельной работы обучающихся: – читальные залы; – учебные залы вычислительной техники А-401, А-509, А-513, А-516, Д-501, Д-503, Д-505, Д-507; – помещения для хранения и профилактического обслуживания учебного оборудования – А-521

<b>8 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ</b>	
Вид учебной деятельности	Организация учебной деятельности обучающегося
Лекция	<p>Лекция (от латинского «lectio» – чтение) – вид аудиторных учебных занятий. Лекция: закладывает основы научных знаний в систематизированной, последовательной, обобщенной форме; раскрывает состояние и перспективы развития соответствующей области науки и техники; концентрирует внимание обучающихся на наиболее сложных, узловых вопросах; стимулирует познавательную активность обучающихся.</p> <p>Во время лекционных занятий обучающийся должен уметь сконцентрировать внимание на изучаемых проблемах и включить в работу все виды памяти: словесную, образную и моторно-двигательную. Для этого весь материал, излагаемый преподавателем, обучающемуся необходимо конспектировать. На полях конспекта следует пометить вопросы, выделенные обучающимся для консультации с преподавателем. Выводы, полученные в виде формул, рекомендуется в конспекте подчеркивать или обводить рамкой, чтобы лучше запоминались. Полезно составить краткий справочник, содержащий определения важнейших понятий лекции. К каждому занятию следует разобрать материал предыдущей лекции. Изучая материал по учебнику или конспекту лекций, следует переходить к следующему вопросу только в том случае, когда хорошо усвоен предыдущий вопрос. Ряд вопросов дисциплины может быть вынесен на самостоятельное изучение. Такое задание требует оперативного выполнения. В конспекте лекций необходимо оставить место для освещения упомянутых вопросов. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, то необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии</p>
Лабораторная работа	Основной целью лабораторных работ является теоретическое обоснование, наглядное и/или экспериментальное подтверждение и/или проверка существенных теоретических положений (законов, закономерностей) анализ существующих методик и

	<p>методов их реализации и т.д. Они занимают преимущественное место при изучении дисциплин обязательной части и части, формируемой участниками образовательных отношений Блока 1.</p> <p>Исходя из цели, содержанием лабораторных работ могут быть:</p> <ul style="list-style-type: none"> <li>- экспериментальная проверка формул, методик расчета;</li> <li>- проведение натуральных измерений свойств, рабочих параметров, режимов работы при помощи лабораторного оборудования и/или стендов и макетов;</li> <li>- ознакомление, анализ и теоретические выкладки по устройству, принципу действия и способам обслуживания аппаратов, деталей машин, механизмов, процессов, протекающих в них при этом и т.д.;</li> <li>- наглядная графическая интерпретация чертежей, схем, объемных поверхностей и т.д., воспроизводимых с помощью специализированного программного обеспечения;</li> <li>- имитационное моделирование процессов, протекающих в сложных химических, физических, механических, электрических и пр. объектах;</li> <li>- наглядное представление о работе персонала конкретной организации или подразделения ОАО «РЖД» посредством моделирования штатных и внештатных ситуаций в виртуальных специализированных АРМ (автоматизированных рабочих мест);</li> <li>- установление и подтверждение закономерностей (путем сравнения проведенного эксперимента и рассчитанных значений) и т.д.;</li> <li>- ознакомление с методиками проведения экспериментов, наглядным устройством стенд-макетов и пр.;</li> <li>- установление свойств веществ, их качественных и количественных характеристик;</li> <li>- анализ различных характеристик процессов, в том числе производственных и иных процессов;</li> <li>- расчет параметров различных явлений и процессов, смоделировать которые не возможно в реальных условиях (например, чрезвычайные ситуации и пр.);</li> <li>- наблюдение развития явлений, процессов и др.</li> </ul> <p>Допускается иное содержание лабораторных работ, если это будет способствовать реализации целей и задач дисциплины и формированию соответствующих компетенций.</p> <p>По характеру выполняемых лабораторных работ возможны:</p> <ul style="list-style-type: none"> <li>- ознакомительные работы, используемые для закрепления изученного теоретического материала;</li> <li>- аналитические работы, используемые для получения новой информации на основе формализованных методов;</li> <li>- творческие работы, ориентированные на самостоятельный выбор подходов решения задач.</li> </ul> <p>Прежде, чем приступить к лабораторным занятиям, обучающимся необходимо повторить теоретический материал по теме работы. Каждая лабораторная работа оснащена методическими указаниями, разработанными преподавателями, ведущими дисциплину</p>
Самостоятельная работа	<p>Обучение по дисциплине «Методология программной инженерии» предусматривает активную самостоятельную работу обучающегося. В разделе 4 рабочей программы, который называется «Структура и содержание дисциплины», все часы самостоятельной работы расписаны по темам и вопросам, а также указана необходимая учебная литература: обучающийся изучает учебный материал, разбирает примеры и решает разноуровневые задачи в рамках выполнения как общих домашних заданий, так и индивидуальных домашних заданий (ИДЗ) и других видов работ, предусмотренных рабочей программой дисциплины. При выполнении домашних заданий обучающемуся следует обратиться к задачам, решенным на предыдущих практических занятиях, решенным домашним работам, а также к примерам, приводимым лектором. Если этого будет недостаточно для выполнения всей работы можно дополнительно воспользоваться учебными пособиями, приведенными в разделе 6.1 «Учебная литература». Если, несмотря на изученный материал, задание выполнить не удастся, то в обязательном порядке необходимо посетить консультацию преподавателя, ведущего практические занятия, и/или консультацию лектора.</p> <p>Домашние задания, индивидуальные домашние задания и другие работы, предусмотренные рабочей программой дисциплины должны быть выполнены обучающимся в установленные преподавателем сроки в соответствии с требованиями к оформлению текстовой и графической документации, сформулированным в Положении «Требования к оформлению текстовой и графической документации. Нормоконтроль»</p>
Комплекс учебно-методических материалов по всем видам учебной деятельности, предусмотренным рабочей программой дисциплины (модуля), размещен в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет	



# **Приложение № 1 к рабочей программе**

## **ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**для проведения текущего контроля успеваемости  
и промежуточной аттестации**

## 1. Общие положения

Фонд оценочных средств (ФОС) является составной частью нормативно-методического обеспечения системы оценки качества освоения обучающимися образовательной программы.

Фонд оценочных средств предназначен для использования обучающимися, преподавателями, администрацией ИрГУПС, а также сторонними образовательными организациями для оценивания качества освоения образовательной программы и уровня сформированности компетенций у обучающихся.

Задачами ФОС являются:

- оценка достижений обучающихся в процессе изучения дисциплины;
- обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения в образовательный процесс;
- самоподготовка и самоконтроль обучающихся в процессе обучения.

Фонд оценочных средств сформирован на основе ключевых принципов оценивания: валидность, надежность, объективность, эффективность.

Для оценки уровня сформированности компетенций используется трехуровневая система:

- минимальный уровень освоения, обязательный для всех обучающихся по завершению освоения образовательной программы; дает общее представление о виде деятельности, основных закономерностях функционирования объектов профессиональной деятельности, методов и алгоритмов решения практических задач;

- базовый уровень освоения, превышение минимальных характеристик сформированности компетенций; позволяет решать типовые задачи, принимать профессиональные и управленческие решения по известным алгоритмам, правилам и методикам;

- высокий уровень освоения, максимально возможная выраженность характеристик компетенций; предполагает готовность решать практические задачи повышенной сложности, нетиповые задачи, принимать профессиональные и управленческие решения в условиях неполной определенности, при недостаточном документальном, нормативном и методическом обеспечении.

## 2. Перечень компетенций, в формировании которых участвует дисциплина.

### Программа контрольно-оценочных мероприятий. Показатели оценивания компетенций, критерии оценки

Дисциплина «Методология программной инженерии» участвует в формировании компетенций:

ОПК-1. Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте

ОПК-3. Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями

УК-6. Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки

#### Программа контрольно-оценочных мероприятий очная форма обучения

№	Наименование контрольно-оценочного мероприятия	Объект контроля	Код индикатора достижения компетенции	Наименование оценочного средства (форма проведения*)
<b>2 семестр</b>				
<b>1.0</b>	<b>Раздел 1 Введение в дисциплину</b>			
1.1	Текущий контроль	Тема 1. Понятие и задачи программной инженерии	ОПК-1.1 ОПК-1.3 ОПК-3.2 УК-6.1 УК-6.2 УК-6.3	Лабораторная работа (письменно/устно)
1.2	Текущий контроль	Тема 2. Проектирование программного обеспечения	ОПК-1.2 ОПК-3.2	Лабораторная работа (письменно/устно)
1.3	Текущий контроль	Тема 3. Жизненный цикл ПО от технического задания на разработку до завершения эксплуатации	ОПК-1.2 ОПК-3.1 ОПК-3.2 ОПК-3.3 УК-6.2	Лабораторная работа (письменно/устно)
<b>2.0</b>	<b>Раздел 2 Стандарты разработки, технико-экономическое обоснование и требования к ПО. Планирование жизненного цикла. Проектирование ПО</b>			
2.1	Текущий контроль	Тема 4. Стандарты и профили стандартов	ОПК-3.1 ОПК-3.2	Доклад (устно)
2.2	Текущий контроль	Тема 5. Системное проектирование приложений	ОПК-1.1 ОПК-1.2 УК-6.2	Лабораторная работа (письменно/устно)
2.3	Текущий контроль	Тема 6. Обоснование разработки ПО	ОПК-1.1 ОПК-3.2	Доклад (устно)
2.4	Текущий контроль	Тема 7. Модели жизненного цикла и их особенности	ОПК-3.1	Лабораторная работа (письменно/устно)
2.5	Текущий контроль	Тема 8. Применение визуального языка моделирования	ОПК-1.1 ОПК-1.3	Лабораторная работа (письменно/устно)
	Промежуточная аттестация	Раздел 1 Введение в дисциплину. Раздел 2 Стандарты разработки, технико-экономическое обоснование и требования к ПО. Планирование жизненного цикла. Проектирование ПО.		Зачет (собеседование) Зачет - тестирование (компьютерные технологии)
<b>3 семестр</b>				
<b>3.0</b>	<b>Раздел 3 Проблемы разработки. Тестирование и документирование ПО</b>			
3.1	Текущий контроль	Тема 9. Ресурсы ПО и	ОПК-1.2	Лабораторная работа

		управление ими	ОПК-3.2 УК-6.2 УК-6.3	(письменно/устно)
3.2	Текущий контроль	Тема 10. Риски разработки ПО. Отладка ПО	ОПК-1.2 ОПК-3.2	Лабораторная работа (письменно/устно)
3.3	Текущий контроль	Тема 11. Верификация программных кодов	ОПК-1.1 ОПК-1.2	Лабораторная работа (письменно/устно)
3.4	Текущий контроль	Тема 12. Сопровождение и мониторинг ПО	ОПК-3.2	Доклад (устно)
3.5	Текущий контроль	Тема 13. Документирование ПО	ОПК-3.2 ОПК-3.3	Лабораторная работа (письменно/устно)
	Промежуточная аттестация	Раздел 1 Введение в дисциплину. Раздел 2 Стандарты разработки, технико-экономическое обоснование и требования к ПО. Планирование жизненного цикла. Проектирование ПО. Раздел 3 Проблемы разработки. Тестирование и документирование ПО.	ОПК-1.1 ОПК-1.2 ОПК-3.1 ОПК-3.3 УК-6.2 УК-6.3	Курсовая работа (письменно) Курсовая работа (устно)
	Промежуточная аттестация	Раздел 1 Введение в дисциплину. Раздел 2 Стандарты разработки, технико-экономическое обоснование и требования к ПО. Планирование жизненного цикла. Проектирование ПО. Раздел 3 Проблемы разработки. Тестирование и документирование ПО.	ОПК-1.1 ОПК-1.2 ОПК-1.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 УК-6.1 УК-6.2 УК-6.3	Экзамен (собеседование) Экзамен - тестирование (компьютерные технологии)

\*Форма проведения контрольно-оценочного мероприятия: устно, письменно, компьютерные технологии.

### **Описание показателей и критериев оценивания компетенций. Описание шкал оценивания**

Контроль качества освоения дисциплины включает в себя текущий контроль успеваемости и промежуточную аттестацию. Текущий контроль успеваемости и промежуточная аттестация обучающихся проводятся в целях установления соответствия достижений обучающихся поэтапным требованиям образовательной программы к результатам обучения и формирования компетенций.

Текущий контроль успеваемости – основной вид систематической проверки знаний, умений, навыков обучающихся. Задача текущего контроля – оперативное и регулярное управление учебной деятельностью обучающихся на основе обратной связи и корректировки. Результаты оценивания учитываются в виде средней оценки при проведении промежуточной аттестации.

Для оценивания результатов обучения используется четырехбалльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» и двухбалльная шкала: «зачтено», «не зачтено».

Перечень оценочных средств, используемых для оценивания компетенций, а также краткая характеристика этих средств приведены в таблице.

#### Текущий контроль

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Доклад	Продукт самостоятельной работы обучающегося, представляющий собой публичное выступление по	Темы докладов

		представлению полученных результатов решения определенной учебно-практической, учебно-исследовательской или научной темы. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	
2	Лабораторная работа	Средство, позволяющее оценить умение обучающегося письменно/устно излагать суть поставленной задачи, самостоятельно применять стандартные методы решения поставленной задачи с использованием имеющейся лабораторной базы, проводить анализ полученного результата работы. Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся	Образец задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

### Промежуточная аттестация

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Зачет	Средство, позволяющее оценить знания, умения, навыков и (или) опыта деятельности обучающегося по дисциплине. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Перечень теоретических вопросов и практических заданий к зачету
2	Тест – промежуточная аттестация в форме зачета	Система автоматизированного контроля освоения компетенций (части компетенций) обучающимся по дисциплине (модулю) с использованием информационно-коммуникационных технологий. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Фонд тестовых заданий
3	Экзамен	Средство, позволяющее оценить знания, умения, навыков и (или) опыта деятельности обучающегося по дисциплине. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Перечень теоретических вопросов и практических заданий (образец экзаменационного билета) к экзамену
4	Тест – промежуточная аттестация в форме экзамена	Система автоматизированного контроля освоения компетенций (части компетенций) обучающимся по дисциплине (модулю) с использованием информационно-коммуникационных технологий. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Фонд тестовых заданий
5	Курсовая работа	Позволяет оценить умения обучающихся самостоятельно конструировать свои знания в процессе решения практических задач и проблем, ориентироваться в информационном пространстве и уровень сформированности аналитических, исследовательских навыков, навыков практического и творческого мышления. Может выполняться в индивидуальном порядке или группой обучающихся. Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся в предметной или межпредметной областях	Образец задания для выполнения курсовой работы и примерный перечень вопросов для ее защиты

### Критерии и шкалы оценивания компетенций в результате изучения дисциплины при проведении промежуточной аттестации в форме зачета и экзамена. Шкала оценивания уровня освоения компетенций

Шкалы оценивания	Критерии оценивания	Уровень освоения компетенции
------------------	---------------------	------------------------------

«отлично»		Обучающийся правильно ответил на теоретические вопросы. Показал отличные знания в рамках учебного материала. Правильно выполнил практические задания. Показал отличные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на все дополнительные вопросы	Высокий
«хорошо»	«зачтено»	Обучающийся с небольшими неточностями ответил на теоретические вопросы. Показал хорошие знания в рамках учебного материала. С небольшими неточностями выполнил практические задания. Показал хорошие умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на большинство дополнительных вопросов	Базовый
«удовлетворительно»		Обучающийся с существенными неточностями ответил на теоретические вопросы. Показал удовлетворительные знания в рамках учебного материала. С существенными неточностями выполнил практические задания. Показал удовлетворительные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Допустил много неточностей при ответе на дополнительные вопросы	Минимальный
«неудовлетворительно»	«не зачтено»	Обучающийся при ответе на теоретические вопросы и при выполнении практических заданий продемонстрировал недостаточный уровень знаний и умений при решении задач в рамках учебного материала. При ответах на дополнительные вопросы было допущено множество неправильных ответов	Компетенция не сформирована

### Тест – промежуточная аттестация в форме зачета и экзамена

Шкала оценивания		Критерии оценивания	
«отлично»	«зачтено»	Обучающийся верно ответил на 90 – 100 % тестовых заданий при прохождении тестирования	
«хорошо»		Обучающийся верно ответил на 80 – 89 % тестовых заданий при прохождении тестирования	
«удовлетворительно»		Обучающийся верно ответил на 70 – 79 % тестовых заданий при прохождении тестирования	
«неудовлетворительно»	«не зачтено»	Обучающийся верно ответил на 69 % и менее тестовых заданий при прохождении тестирования	

### Курсовая работа

Шкала оценивания	Критерии оценивания
«отлично»	Содержание курсовой работы полностью соответствует заданию. Представлены результаты обзора литературных и иных источников. Структура курсовой работы логически и методически выдержана. Все выводы и предложения убедительно аргументированы. Оформление курсовой работы и полученные результаты полностью отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся правильно и уверенно отвечает на вопросы преподавателя, демонстрирует глубокое знание теоретического материала, способен аргументировать собственные утверждения и выводы
«хорошо»	Содержание курсовой работы полностью соответствует заданию. Представлены результаты обзора литературных и иных источников. Структура курсовой работы логически и методически выдержана. Большинство выводов и предложений аргументировано. Оформление курсовой работы и полученные результаты в целом

	отвечают требованиям, изложенным в методических указаниях. Имеются одна-две несущественные ошибки в использовании терминов, в построенных диаграммах и схемах. Наличествует незначительное количество грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся правильно и уверенно отвечает на большинство вопросов преподавателя, демонстрирует хорошее знание теоретического материала, но не всегда способен аргументировать собственные утверждения и выводы. При наводящих вопросах преподавателя исправляет ошибки в ответе
«удовлетворительно»	Содержание курсовой работы частично не соответствует заданию. Результаты обзора литературных и иных источников представлены недостаточно полно. Есть нарушения в логике изложения материала. Аргументация выводов и предложений слабая или отсутствует. Имеются одно-два существенных отклонений от требований в оформлении курсовой работы. Полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две существенных ошибки в использовании терминов, в построенных диаграммах и схемах. Много грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся допускает грубые ошибки при ответах на вопросы преподавателя и /или не дал ответ более чем на 30% вопросов, демонстрирует слабое знание теоретического материала, в большинстве случаев не способен уверенно аргументировать собственные утверждения и выводы
«неудовлетворительно»	Содержание курсовой работы в целом не соответствует заданию. Имеются более двух существенных отклонений от требований в оформлении курсовой работы. Большое количество существенных ошибок по сути работы, много грамматических и стилистических ошибок и др. Полученные результаты не отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся демонстрирует слабое понимание программного материала. Курсовая работа не представлена преподавателю. Обучающийся не явился на защиту курсовой работы

### **Критерии и шкалы оценивания результатов обучения при проведении текущего контроля успеваемости**

#### Доклад

Шкалы оценивания		Критерии оценивания
«отлично»		Доклад создан с использованием компьютерных технологий (презентация PowerPoint, Flash–презентация, видео-презентация и др.) Используются дополнительные источники информации. Содержание заданной темы раскрыто в полном объеме. Отражена структура доклада (вступление, основная часть, заключение, присутствуют выводы и примеры). Оформление работы. Оригинальность выполнения (работа сделана самостоятельно, представлена впервые)
«хорошо»	«зачтено»	Доклад создан с использованием компьютерных технологий (презентация PowerPoint, Flash–презентация, видео-презентация и др.) Содержание доклада включает в себя информацию из основных источников (методическое пособие), дополнительные источники информации не использовались. Содержание заданной темы раскрыто не в полном объеме. Структура доклада сохранена (вступление, основная часть, заключение, присутствуют выводы и примеры)
«удовлетворительно»		Доклад сделан устно, без использования компьютерных технологий. Содержание доклада ограничено информацией только из методического пособия. Содержание заданной темы раскрыто не в полном объеме. Отсутствуют выводы и примеры. Оригинальность выполнения низкая
«неудовлетворительно»	«не зачтено»	Доклад сделан устно, без использования компьютерных технологий и других наглядных материалов. Содержание ограничено информацией только из методического пособия. Заданная тема доклада не раскрыта, основная мысль доклада не передана

#### Лабораторная работа

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет без замечаний. Лабораторная работа выполнена обучающимся в полном объеме с

		соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; показал необходимые для проведения работы теоретические знания, практические умения и навыки. Работа (отчет) оформлена аккуратно, в наиболее оптимальной для фиксации результатов форме
«хорошо»		Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет с небольшими недочетами. Лабораторная работа выполнена обучающимся в полном объеме и самостоятельно. Допущены отклонения от необходимой последовательности выполнения, не влияющие на правильность конечного результата. Работа показывает знание обучающимся основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допущены неточности и небрежность в оформлении результатов работы (отчета)
«удовлетворительно»		Лабораторная работа выполнена с задержкой, письменный отчет с недочетами. Лабораторная работа выполняется и оформляется обучающимся при посторонней помощи. На выполнение работы затрачивается много времени. Обучающийся показывает знания теоретического материала, но испытывает затруднение при самостоятельной работе с источниками знаний или приборами
«неудовлетворительно»	«не зачтено»	Лабораторная работа не выполнена, письменный отчет не представлен. Результаты, полученные обучающимся, не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Показывается плохое знание теоретического материала и отсутствие необходимых умений. Лабораторная работа не выполнена, у учащегося отсутствуют необходимые для проведения работы теоретические знания, практические умения и навыки

### 3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

#### 3.1 Типовые контрольные темы для написания докладов

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов тем для написания докладов.

#### Образец тем докладов

##### «Тема 4. Стандарты и профили стандартов»

1. Базовый стандарт ГОСТ Р ИСО/МЭК 12207-99
2. Стандарты в области программного обеспечения ИСО/МЭК
3. Стандартизация и метрология в разработке программного обеспечения.
4. Стандартизация информационных технологий
5. Действующие стандарты и проблемы программных интерфейсов.
6. Государственный комитет РФ по стандартизации.

#### Образец тем докладов

##### «Тема 6. Обоснование разработки ПО»

1. Выбор и обоснование критериев эффективности и качества разрабатываемой программы
2. Обоснование необходимости проведения научно-исследовательских работ.
3. Выбор потенциально возможных вариантов для сравнения
4. Расчет затрат на разработку ПО, цены разработки, целесообразных объемов продаж.
5. Выбор метода и расчет экономической эффективности разработки ПО
6. Стоимостная оценка результатов и единовременных затрат на внедрение ПО.
7. Расчет текущих затрат на функционирование системы.



## 8. Выбор метода и оценка экономической эффективности внедрения ПО.

### Образец тем докладов «Тема 12. Сопровождение и мониторинг ПО»

1. Инспектирование.
2. Сопровождение ПО.
3. Свойства сопровождаемого ПО.
4. Реинженерия ПО.
5. Наследуемые (legasy) системы.
6. Повторное использование и переносимость ПО.

### **3.2 Типовые задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты**

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

#### «Тема 1. Понятие и задачи программной инженерии»

Задание: разработать класс для представления комплексных чисел с возможностью задания вещественной и мнимой частей как числами типов `double`, так и целыми числами. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел.

Вопросы:

1. Методологии разработки ПО?
2. Классификация методологий разработки ПО?
3. Системные основы современных технологий программной инженерии?
4. Общая теория системных основ технологий программной инженерии?
5. Технологии разработки ПО?
6. Общая теория и примеры технологий разработки ПО (Microsoft Solution Framework (MSF)?
7. Программный процесс и модель программного процесса. Методы программной инженерии. Что такое CASE системы?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

#### «Тема 2. Проектирование программного обеспечения»

Задание: разработать техническое задание на автоматизацию управления деятельностью предприятия согласно ГОСТ 34.602–89 «Техническое задание на создание (развитие или модернизацию) системы» на примере гипотетического предприятия по выбранной теме.

Примеры предметных областей:

1. Ж/Д вокзал. Учет продажи билетов.
2. Информация в отделе кадров.
3. Учет движения товаров на складе.
4. Гостиница. Размещение клиентов.
5. Банк. Работа с клиентами.
6. Составление расписания занятий.
7. Налоговая инспекция. Учет уплаты налогов.
8. Страховая компания. Заключение договоров.
9. Ведение библиотечного фонда.
10. Городская телефонная сеть. Учет междугородных переговоров.
11. Театр. Продажа билетов.

12. Кадровое агентство.
13. Компьютерный сервисный центр.
14. Риэлтерская фирма.
15. Учет движения квартир

Вопросы:

1. Что входит в основы качества.
2. Определение качества.
3. Характеристики качества.
4. Рассмотрение показателей качества: функциональной пригодности, надежности, практичности, эффективности, сопровождаемости, мобильности.
5. Модели качества процессов конструирования ПО. Качество процессов
6. Описание процессов управления качеством.
7. Методы управления качеством ПО.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Тема 3. Жизненный цикл ПО от технического задания на разработку до завершения эксплуатации»

Ознакомление со схемой классификации проектов по разработке программных средств и систем, предложенной Институтом качества программного обеспечения SQI (Software Quality Institute) для выбора модели жизненного цикла.

Задание: выбрать модель жизненного цикла и обосновать выбор модели ЖЦ для информационной системы гипотетического предприятия на основании четырех категорий критериев:

1. характеристики требований к проекту;
2. характеристики команды разработчиков;
3. характеристики пользователей (заказчиков);
4. характеристики типов проектов и рисков.

Вопросы:

1. Язык UML?
2. Назначение и структура языка UML?
3. Синтаксис и семантика моделей на языке UML?
4. Диаграммы прецедентов, состояний, активности, взаимодействия, классов, компонент, последовательности действий и другие на языке UML?
5. Назначение профилей стандартов жизненного цикла в программной инженерии?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Тема 5. Системное проектирование приложений»

Задание: сделать описание на основе разработанного технического задания:

- структурной схемы программного продукта (схема, отражающая состав и взаимодействие по управлению частями разрабатываемого программного обеспечения);
- функциональной схемы (схема взаимодействия компонентов программного обеспечения с описанием информационных потоков, состава данных в потоках и указанием используемых файлов и устройств);
- интерфейса системы в виде дерева диалога.

Рассмотреть меры оценки характеристик модулей – связности и сцепления – для проверки корректности и эффективности структурного разбиения программы.

Вопросы:

1. Какие факторы объектно-ориентированных систем влияют на метрики для их оценки и как проявляется это влияние?
2. Охарактеризуйте метрики связности классов по данным.
3. Охарактеризуйте метрики связности классов по методам.
4. Какие характеристики объектно-ориентированных систем ухудшает сцепление

классов?

5. Объясните, как определить сцепление классов с помощью метрики «зависимость изменения между классами».

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Тема 7. Модели жизненного цикла и их особенности»

Задание: составить список требований, предъявляемых к ПО. Требования должны делиться на следующие категории:

- функциональные;
- требования пользователей сайта;
- требования владельцев сайта;
- нефункциональные.

Требования необходимо оформить в соответствии с шаблонами RUP (документ SRS - Software Requirements Specification). Для каждого из требований нужно указать его атрибуты (в соответствии с методологией RUP), а также оценить и аргументировать приблизительное количество часов, требующихся на реализацию этого требования.

Указать какая модель жизненного цикла будет использоваться, пояснить выбор.

Отчёт по лабораторной работе должен содержать:

1. Документ Software Requirements Specification, содержащий список требований к сайту.
2. UseCase-диаграммы прецедентов использования, реализующих функциональные требования.
3. Модель жизненного цикла. Выводы по работе.

Вопросы:

1. Что такое инженерия качества?
2. Поясните концепцию инженерии качества?
3. Опишите процессы инженерии качества?
4. Модели жизненного цикла программного обеспечения?
5. Деятельность группы качества на стадиях жизненного цикла?
6. Инструментарий технологии разработки ПО?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Тема 8. Применение визуального языка моделирования»

Задание

1. Построить диаграммы по методике представления на физическом уровне элементов программных систем с помощью диаграмм компонентов UML.
2. Реализовать построения диаграмм компонентов UML в среде Rational Rose по разрабатываемому ПО.

Вопросы:

1. Поясните два подхода к моделированию поведения системы. Объясните достоинства и недостатки каждого из этих подходов.
2. Охарактеризуйте вершины и дуги диаграммы схем состояний. В чем состоит назначение этой диаграммы?
3. Как отображаются действия в состояниях диаграммы схем состояний?
4. Как показываются условные переходы между состояниями?
5. Как задаются вложенные состояния в диаграммах схем состояний?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Тема 9. Ресурсы ПО и управление ими»

Задание: сконфигурировать в каталоге репозитории svn и git и загрузить в них начальную ревизию файлов с исходными кодами (в соответствии с выданным вариантом).

Воспроизвести последовательность команд для систем контроля версий svn и git, осуществляющих операции над исходным кодом, приведённые на блок-схеме.

При составлении последовательности команд необходимо учитывать следующие условия:

- цвет элементов схемы указывает на пользователя, совершившего действие (красный - первый, синий - второй);
- цифры над узлами - номер ревизии. Ревизии создаются последовательно;
- необходимо разрешать конфликты между версиями, если они возникают.

Отчёт по работе должен содержать:

1. Задание и блок-схему в соответствии с реализуемым проектом.
2. Список команд, использованных при создании и конфигурации репозитория в каталоге пользователя.
3. Номера ревизий и соответствующие им последовательности команд с комментариями (для svn и git).
4. Выводы по работе.

Вопросы:

1. Системы контроля версий - назначение, примеры решений.
2. Ревизии и ветки.
3. Основные операции над данными в системах контроля версий. Основные команды svn и git.
4. Виды конфликтов и способы их решения.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Тема 10. Риски разработки ПО. Отладка ПО»

Задание

1. Изучить возможности модульного тестирования в MS Visual Studio средствами NUnit и Unit Testing Framework.
2. Выполнить тестирование заданного модуля программы на языке C#.
3. Провести оценку рисков разработки ПО.

Пример задания. Для уменьшения риска несанкционированного доступа к документом разработки программной системы создать парольную защиту вида: Простой пароль длиной не менее 6 символов, содержащий не менее 1 цифры и буквы в разных регистрах. Трёхкратный неправильный ввод приводит к предупреждению на экране, что за компьютером злоумышленник, и выдаче звукового сигнала длиной 15 сек. После шести запусков программа требует и обязательно производит смену пароля. Пароль хранится в закодированном виде.

Вопросы:

1. Назовите средства автоматизации тестирования ПС на уровне модулей.
2. Дайте сравнительную характеристику возможностей NUnit и Unit Testing Framework.
3. Какие существуют виды тестирования?
4. Какие Вы знаете методики структурного тестирования?
5. Какие Вы знаете методики функционального тестирования?
6. Какие известны подходы тестирования интеграции программной системы?
7. Для чего применяются пароли?
8. Какие рекомендации целесообразно соблюдать при использовании паролей?
9. Как можно реализовать разграничение доступа к программе?
10. Чем отличается логическая бомба от вируса?
11. Что такое НСД?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

### «Тема 11. Верификация программных кодов»

Задание: рассмотреть различные методы тестирования и отработки навыков составления наборов тестов для проверки функциональности и повышения уровня надежности программных средств.

Вопросы:

1. Методы и средства тестирования разрабатываемого ПО?
2. Нагрузочное тестирование и тестирование производительности?
3. Модульное тестирование. Интеграционное тестирование?
4. Системное тестирование?
5. Альфатестирование?
6. Бета-тестирование?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

### «Тема 13. Документирование ПО»

Задание: разработать документацию на регистрацию прав на программный продукт. Оформление документации для получения свидетельства о государственной регистрации программы для ЭВМ.

Вопросы:

1. Содержание документов по ПО
2. Документация, создаваемая в процессе разработки программных средств. Документы управления разработкой ПС. Документы, входящие в состав ПС
3. Пользовательская документация. Помощь в освоении ПО

### 3.3 Типовые контрольные задания для проведения тестирования

Фонд тестовых заданий по дисциплине содержит тестовые задания, распределенные по разделам и темам, с указанием их количества и типа.

#### Структура фонда тестовых заданий по дисциплине

Индикатор достижения компетенции	Тема в соответствии с РПД	Характеристика ТЗ	Количество тестовых заданий, типы ТЗ
ОПК-1.1 ОПК-1.3 ОПК-3.2	Тема 1. Понятие и задачи программной инженерии	Знание	5 – ОТЗ 5 – ЗТЗ
ОПК-1.2 ОПК-3.2	Тема 2. Проектирование программного обеспечения	Знание	4 – ОТЗ 6 – ЗТЗ
		Умение	2 – ОТЗ 2 – ЗТЗ
		Навык и (или) опыт деятельности/действие	2 – ОТЗ 2 – ЗТЗ
ОПК-1.2 ОПК-3.1 ОПК-3.2 ОПК-3.3	Тема 3. Жизненный цикл ПО от технического задания на разработку до завершения эксплуатации	Знание	2 – ОТЗ 2 – ЗТЗ
		Умение	2 – ОТЗ 2 – ЗТЗ
		Навык и (или) опыт деятельности/действие	2 – ОТЗ 2 – ЗТЗ
ОПК-3.1 ОПК-3.2	Тема 4. Стандарты и профили стандартов	Знание	4 – ОТЗ 4 – ЗТЗ
ОПК-1.1 ОПК-1.2	Тема 5. Системное проектирование приложений	Знание	2 – ОТЗ 2 – ЗТЗ
		Умение	4 – ОТЗ 4 – ЗТЗ
ОПК-1.1 ОПК-3.2	Тема 6. Обоснование разработки ПО	Знание	6 – ОТЗ 6 – ЗТЗ

		Умение	2 – ОТЗ 2 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	2 – ОТЗ 2 – ЗТЗ
ОПК-3.1	Тема 7. Модели жизненного цикла и их особенности	Знание	4 – ОТЗ 4– ЗТЗ
ОПК-1.1 ОПК-1.3	Тема 8. Применение визуального языка моделирования	Знание	4 – ОТЗ 4 – ЗТЗ
		Умение	2 – ОТЗ 2 – ЗТЗ
ОПК-1.2 ОПК-3.2	Тема 9. Ресурсы ПО и управление ими	Знание	2 – ОТЗ 2 – ЗТЗ
ОПК-1.2 ОПК-3.2	Тема 10. Риски разработки ПО. Отладка ПО	Знание	2 – ОТЗ 2 – ЗТЗ
ОПК-1.1 ОПК-1.2	Тема 11. Верификация программных кодов	Знание	2 – ОТЗ 2 – ЗТЗ
ОПК-3.2	Тема 12. Сопровождение и мониторинг ПО	Знание	2 – ОТЗ 2 – ЗТЗ
ОПК-3.2 ОПК-3.3	Тема 13. Документирование ПО	Знание	2 – ОТЗ 2 – ЗТЗ
		Итого	120

Полный комплект ФТЗ хранится в электронной информационно-образовательной среде ИрГУПС и обучающийся имеет возможность ознакомиться с демонстрационным вариантом ФТЗ.

Ниже приведен образец типового варианта итогового теста, предусмотренного рабочей программой дисциплины.

1. Легкость применения программного обеспечения это:

- а) **характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке исходных данных, применению ПО;**
- б) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов;
- в) характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации.

2. Мобильность программного обеспечения это:

- а) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей;
- б) способность ПС безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени;
- в) **способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое.**

3. В жизненном цикле программной системы третьей является фаза....

Ответ: программирование

4. Устойчивость программного обеспечения — это:

- а) свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные;
- б) **свойство, способна противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя;**
- в) свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на неправильные (ошибочные) входные данные.

5. Означает сочетание структур данных с методами их обработки в абстрактных типах данных?

Ответ: инкапсуляция

6. При конструировании программного обеспечения на этапе разработки или выбора алгоритма решения реализуется следующее:

- а) архитектурное обработки программы;
- б) выбор языка программирования;**
- в) совершенствование программы.

7. Абстрактный образ или ситуация?

Ответ: фреймы

8. На этапе тестирования пользователь выполняет следующее:

- а) синтаксическое отладки;
- б) выбор тестов и метода тестирования;**
- в) определение формы выдачи результатов.

9. Как называется проверка правильности трансформации проекта в программу?

Ответ: верификация

10. Архитектура программной системы —

- а) декомпозиция решения для выделенного спектра задач домена на подсистемы или иерархию подсистем;
- б) определение системы в терминах вычислительных составляющих (подсистем) и интерфейсов между ними, которое отражает правила декомпозиции проблемы на составляющие;**
- в) соответствующие вариации состава выделенных компонент.

11. Унифицированный язык визуального моделирования, использует нотацию Диаграмм

Ответ: UML

12. Проектирование ПО в основном рассматривается как

- а) архитектурное проектирование;**
- б) коммуникационные методы;
- в) детальные методы.

13. Набор методов взаимодействия компьютерной программы и пользователя этой Программ

Ответ: интерфейс

14. Модель жизненного цикла —

- а) определение определенных действий, которые сопровождают изменения состояний объектов;
- б) типичная схема последовательности работ на этапах разработки программного продукта;**
- в) отражение динамики изменений состояния каждого класса объектов.

15. Как называется процесс обеспечения соответствия разработки требованиям ее заказчиков.

Ответ: Валидация

16. Как называется процесс разбиения одной сложной задачи на несколько простых подзадач?

Ответ: декомпозиция.

17. Любой продукт деятельности специалистов по разработке программного обеспечения

Ответ: артефакт

18. Объединение нескольких понятий в новое понятие, существенные признаки нового понятия при этом могут быть либо суммой компонент или существенно новыми (отношение «доля — целое»).

Ответ: агрегация.

### 3.4 Типовое задание для выполнения курсовой работы

Типовые задания выложены в электронной информационно-образовательной среде ИргУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец задания для выполнения курсовой работы и примерный перечень вопросов для ее защиты.

Образец типового задания для выполнения курсовой работы

При выполнении курсовой работы обучающиеся должны приобрести практические навыки проектирования и разработки программных продуктов. Для разработки программного продукта можно использовать любой из представленных языков программирования C#, Delphi, Java, JavaScript, Python, PHP, Unity, с применением архитектура территориально-распределенных хранилищ данных, СУБД Oracle, СУБД PostgreSQL, тестирование программного обеспечения, процессы и средства верификации и тестирования программных компонентов.

Задание на курсовую работу

1. Провести анализ предметной области, включая анализ существующих систем.
2. Обосновать выбор технологий для решения поставленных задач.
3. Выполнить проектирования программного обеспечения.

Содержание

Введение (1,5 - 2 стр.)

#### 1. ПРЕДПРОЕКТНОЕ ОБСЛЕДОВАНИЕ

- 1.1. Анализ предметной области и объекта исследования
- 1.2. Постановка задачи (глазами пользователя)
- 1.3. Входные и выходные документы (данные)
- 1.4. Анализ бизнес-процессов «объекта исследования» (макс. 5 стр.)
- 1.5. Информационные технологии разработки ИС

#### 2. ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ

- 2.1. Построение интегрированной бизнес-модели «объекта исследования» (макс. 8 стр.)
- 2.2. Построение инфологической модели ИС (IDEF1X) (макс. 5 стр.)
- 2.3. Требования к разрабатываемой ИС (макс. 5 стр.)
  - 2.3.1. Концептуальные требования к функциональности ИС
  - 2.3.2. Структура ИС и назначение каждого блока (модуля)
  - 2.3.3. Исходные и результирующие документы ИС

#### 3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ИС «НазваниеИС»

- 3.1. Обоснование выбора ПО для разработки (макс. 2 стр.)
- 3.2. Структура ИС и функционирование каждого блока (макс. 5 стр.)
- 3.3. Структура базы данных (макс. 3 стр.)
- 3.4. Распределение ролей пользователей ИС (макс. 2 стр.)
- 3.5. Результаты работы ИС (макс. 2 стр.)
- 3.6. Оценка экономической эффективности внедрения продукта (макс. 2 стр.)

Заключение (1,5 – 3 стр.)

Литература

Приложения

**Введение.**



В начале введения обычно присутствует небольшая аннотация к проблематике курсовой работы.

Во введении обязательно должны содержаться: Объект исследования, Предмет исследования, Цель исследования и задачи, которые необходимо решить для достижения Цели. Задачи соответствуют названиям пунктов содержания (более развернуто и понятно).

Далее следует краткое описание структуры работы по частям, и краткое описание материала, методов и средств, выводов по каждой части.

#### **Объект исследования.**

1. Реальный социально-экономический объект: организация, предприятие, его подразделение, отдел, цех и т.д. При этом если Ваша ИС разрабатывается для подразделения/отдела, то именно этот отдел и является Объектом исследования, а не вся организация. В этом случае Ваша ИС должна полностью охватывать ВСЕ бизнес-процессы на предприятии/организации/подразделении/отделе.

2. Некоторый бизнес-процесс на предприятии, организации, подразделении или отделе. При этом Объектом исследования будет этот самый бизнес-процесс на организации/подразделении/отделе, а не организация/подразделение/отдел.

**Предметом исследования** являются методы и средства инженерии программного обеспечения, применяемые Вами для разработки ИС. Если разрабатываемая Вами ИС служит для автоматизации некоего бизнес-процесса, то Предметом исследования должен быть не сам бизнес-процесс, а методы и средства разработки ИС для автоматизации этого бизнес-процесса.

**Целью исследования** является разработка прототипа информационной системы или программного комплекса для соответствующего бизнес-процесса или организации. В случае, если в работе нет реализации прикладного ПО, целью исследования является разработка проекта информационной системы или программного комплекса для соответствующего бизнес-процесса или организации.

#### **Заключение.**

Напоминается цель исследования. Проводится анализ выполненных задач по частям с соответствующими выводами. Указывается степень достижения цели исследования. Также указываются возможные дальнейшие этапы разработки, внедрения, перспективы и т.д.

#### **Построение интегрированной бизнес-модели «объекта исследования»**

По стандартам IDEF:

- Диаграммы организационной структуры
- Функциональная модель ИС (IDEF0)
- Модель потоков данных (IDEF1)
- Процессная модель (IDEF3)
- Диаграммы ролей пользователей ИС (Swimmer Lane)

Либо по стандартам UML:

- Диаграммы использования ИС
- Диаграммы классов
- Диаграммы состояния
- Иерархия классов и форм

Пример выполненного задания

#### 1 Анализ предметной области

##### 1.1 Создания и эксплуатация цифровых сервисов

Цифровой сервис – самостоятельный продукт информационной деятельности, способствующий реализации бизнес-процессов, их трансформации в цифровую форму на основе применения информационных технологий и каналов цифровой коммуникации, это средство предоставления ценности заказчику (как самой организации, так и внешних пользователей), которую он получает с использованием информационных технологий, не неся при этом рисков и затрат, связанных с обслуживанием сервиса.

Экономический смысл создания и эксплуатации цифровых сервисов описан в следующем определении: цифровой сервис представляет собой комплекс услуг, который обеспечивает пользователю возможность удаленной работы с информационными ресурсами,

не наделяя его правом собственности на эти ресурсы и по договоренности с владельцем ресурса оплачивающим услугу либо пользуясь услугой бесплатно, однако при этом, создавая для владельца сервиса некие позитивные эффекты

При определении понятия цифрового сервиса, например, позиция М. Schrage заключается в том, что текущий механизм является частью процесса виртуализации управления организациями, представляет собой информационные ресурсы, созданные для удовлетворения конкретных нужд пользователя, представляющих ценность, поскольку он в состоянии эти потребности удовлетворить. Приведенное определение дает основание говорить о том, что существует рынок цифровых сервисов, на котором происходит не только их обращение, но и формируются условия конкуренции в плане способностей аналогичных сервисов наиболее полно удовлетворять потребности клиентов.

Другими словами, цифровой сервис представлен любым, по своей форме распространения программным продуктом, будь то интернет-магазин, десктопное приложение управлением внутренней системой корпорации, чат-бот для мессенджера, облачное хранилище, высоконагруженная система управления бизнес-процессами и многое другое.

Можно выделить следующие атрибуты цифрового сервиса:

- способ доступа (доступ удаленный);
- способ использования (по усмотрению заказчика);
- интеграция (может быть полностью интегрирован в деятельность организации);
- ценность (удовлетворение конкретной потребности заказчика);
- совместимость (способность работать с иными сервисами, используемыми в организации);
- условия доступа (условия доступа должны отвечать возможностям клиента).

Говоря об атрибутах, имеется в виду атрибутивная система, функционирование которой зависит от наличия набора характерных свойств, объединённых понятием «цифровой сервис». Поэтому каждый атрибут может иметь вариативность признаков, характеризующих его, например, «способ доступа» может характеризовать сервис как раздвоенность «доступный» - «недоступный», таким образом, речь идёт об удобстве сервиса с точки зрения возможности его реального использования. Поэтому при описании цифрового сервиса можно применять описанную атрибутивную систему, предложив множество признаков [2].

Трансформация экономики в цифровую и разработка цифровых сервисов требуют уточнения и переосмысления коммуникационных, управленческих, образовательных и других подходов, распространенных на данный момент. При переходе процессов управления на использование цифровых сервисов имеет смысл и целесообразно использовать знания и опыт управления сектора услуг информационных технологий (ИТ). В этой сфере накоплен серьезный опыт сервисного управления в условиях предоставления заказчику сервисов, ценность которых базируется на используемых информационных технологиях и архитектурном подходе, главенствующим в котором является объектно-ориентированный подход.

Распространенным примером цифрового сервиса является система хранения, обработки и предоставления сравнительных таблиц и отчетов по результатам деятельности кампании. Отметим, что когда речь идет о разработке подобного сервиса, предоставляющего возможность вычислений, нужно говорить о конструировании такой информационной системы, которая является для функционального заказчика востребованной, соответствующей предъявляемым требованиям, удобной в использовании, адаптивной к потребностям предприятия, универсальной с точки зрения интеграции в деятельность организации, и, скорее всего, доступной в любое время. Такая онтологическая цепочка создает предпосылки для того, чтобы с одной стороны определять цели, которых следует достичь, проектируя цифровой сервис, а с другой стороны – оценивать различные цифровые сервисы по уровню их конкурентоспособности. В то же время под «созданием и эксплуатацией цифрового сервиса», необходимо понимать, что такая деятельность всегда ориентирована на экономический результат, который является воплощением, с одной

стороны, неосвязаемости сервисной услуги, а с точки зрения заказчика фундаментальной причиной оплаты услуг на разработку программного продукта.

Экономический смысл создания и эксплуатации цифровых сервисов описан в следующем определении: цифровой сервис представляет собой комплекс услуг, который обеспечивает пользователю возможность удаленной работы с информационными ресурсами, не наделяя его правом собственности на эти ресурсы и по договоренности с владельцем ресурса оплачивающим услугу либо пользуясь услугой бесплатно, однако при этом, создавая для владельца сервиса некие позитивные эффекты [3]. В таком определении обращает на себя внимание экономическая составляющая использования сервиса как ценности, т. е. возможности и варианты его использования устанавливает непосредственно владелец сервиса, однако в любом случае пользование сервисом несет в себе экономический смысл для всех участников информационного обмена, и заключается он, прежде всего, в экономии ресурсов предприятия, чаще всего, в долгосрочной перспективе, а также нередкой причиной является снижение влияния человеческого фактора на качество выполняемых бизнес-процессов кампании.

Таким образом, цифровой сервис является необходимой составляющей деятельности любой кампании, нацеленной на конкурентоспособную деятельность в целевом секторе рынка. Желаемыми результатами являются: снижение рисков, снижение издержек, снижение влияние составляющей человеческого ресурса на производственные цепочки предприятия, повышение производительности труда и экономия ресурсов. Проводя аналогию с темой настоящей работы, целью является снижение расходов ресурсов на проведение профориентационных работ среди школьников и абитуриентов.

### 1.1 Исследование процессов проведения профориентационных работ школьникам

Проведя анализ предметной области, были выявлены следующие наиболее используемые формы профориентационных работ [4, 5]:

- встречи со специалистами;
- беседы о профессиях;
- дни открытых дверей в вузах;
- различные детские и юношеские лагеря, клубы, кружки, летние школы;
- проведение олимпиад школьников, конкурсов, викторин, фестивалей, игр и квестов;
- консультация для заключения о профессиональной пригодности с представителями школы, мастерами производственного обучения;
- издаваемая в вузах справочная литература: книги, рекламные проспекты, плакаты, фотоальбомы, в которых рассказывается об истории вуза, о содержании профессии, о жизни и быте студентов;
- создание в университетах работниками телевидения и педагогами профориентационных фильмов.

Одной из наиболее предпочтительных форм профориентации является игровая. Профориентационные игры помогают преодолеть недостатки профконсультаций. Игры, используемые в профориентации, включают в себя систему практических заданий и проблемных ситуаций, стимулирующих у школьников анализ ситуации профессионального самоопределения.

Профориентационные игры сводят к минимуму влияние испытующего, позволяют испытуемому раскрыться и подогревают интерес к процессу.

При исследовании профориентационной деятельности были рассмотрены несколько научных работ, в которых в той или иной мере приводятся аргументы о важности развития и проведения работ в этой деятельности, а также предлагаются свои методы и пути развития.

В статье «Профориентация на базе интернет-технологий: просто, доступно, продуктивно» автор Саломатов А. В. пишет, что информационные ресурсы могут существенно облегчить профориентационную работу педагогов и методистов и приводит пример информационного ресурса профориентационного комплекса «Эстафета поколений».

Автором были выделены следующие плюсы внедрения информационного комплекса в профориентационную деятельность:

- комплекс представляет простой способ анализа интересов и достижений подростка;
- зарегистрированные преподаватели могут отслеживать в автоматическом режиме профессиональные предпочтения и склонности детей. Основываясь на этих данных, гораздо проще разработать программу профориентационных мероприятий для выбранной группы учащихся;
- упомянутые выше анкеты школьников позволят сотрудникам профессиональных образовательных организаций ознакомиться с интересами и достижениями старшеклассников, подобрать кандидатов для будущей учебы, разослать школьникам, их родителям и педагогам приглашения на свои образовательные программы и мероприятия.

Подводя итог, автор сделал вывод, что внедрение информационных технологий в сектор профессиональной ориентации детей и молодежи вкупе с действующими механизмами позволит добиться реального социально-экономического эффекта, сделать процесс профориентации для всех его участников проще, интереснее и привлекательнее [6].

В статье «Цифровые сервисы профориентации: возможности и риски» Громова Е. М., Беркутов Д. И. и Горшкова Т. А. рассматривают и сравнивают, оказываемые обучающимся и их родителям различные профориентационные услуги. Также авторы в рамках проекта «Вектор профессионального становления» выполнили работы по апробации цифровых сервисов профориентации. Использовали ресурс «Психологические тесты онлайн» и на собственном опыте подтвердили удобство применения данного цифрового ресурса. Авторы пишут, что указанный профориентационный цифровой сервис был выбран из-за простого интерфейса и навигации, доступными для восприятия школьниками.

По результатам проведённых работ, авторы отмечают, что апробированный ими в ходе экспериментальной работы цифровой сервис «Психологические тесты онлайн» позволяет оперативно осуществлять профориентационную диагностику, является доступным для респондентов и удобным для профконсультантов. Однако при всех положительных моментах цифровизации профориентационной работы использование диагностических средств несёт и определённые риски, связанные с необходимостью грамотной интерпретации результатов, пройденных школьниками диагностик [7].

В статье «Разработка программного проекта информационного сервиса профориентации абитуриентов СФУ» Безруких А. Д., Черепанов М. Д., Мельников В. А. и Мельникова Е.В. описывают разработку проекта информационного сервиса профориентации абитуриентов с использованием графического языка UML, методологии SADT и стандартов IDEF. Сервис позволяет проводить профессиональную диагностику абитуриентов и поддерживать принятие решений.

В результате работы в статье рассмотрен процесс разработки программного проекта и предложено дальнейшее развитие сервиса в направлении интеграции в цифровую платформу, позволяющую как выполнять профориентационную функцию, так и выстраивать образовательную траекторию обучающихся, управлять талантами, учитывая запрос компетенций в масштабах региона [8].

В статье «Разработка профориентационной игры «Лесная промышленность»» авторами Товбис Е. М. и Лис Е. В. обсуждается проблема разработки новых форм профессиональной ориентации абитуриентов и описывается процесс разработки профориентационной игры, сформулированы правила игры и приведён результат тестовой эксплуатации. Анализ показал, что выбор около 75% абитуриентов, поступивших в вуз, совпал с рекомендациями, выданными системой.

В заключении авторами был сделан вывод, что в настоящий момент в связи с малой эффективностью очных мероприятий и ориентированностью молодежи на электронные формы общения возникла необходимость в разработке новых, заочных видов профориентации. Одним из самых желательных видов при этом является профессиональная ориентация в форме игры. Таким образом, предложен новый способ профессиональной

ориентации школьников и абитуриентов – в виде компьютерной профориентационной игры. [9]

В результате анализа научных работ, можно сделать вывод, что профориентационные работы являются актуальным вопросом для всех сфер образования, а грамотное применение информационных технологий позволит повысить эффективность и качество профориентации.

Исследовав процессы проведения профориентации школьников, были выявлены следующие недочёты при использовании любой из форм профориентационных работ:

- выделение места, человеческих ресурсов для проведения профориентационных работ;
- ошибки при проверке результатов профориентации;
- невозможность быстро выдать результаты профориентации;
- сложность соблюдения временных ограничений по проведению профориентации;
- сложность анализировать результаты за все проведённые профориентационные работы.

Решением выявленных недочётов является создание базы данных (БД) приложения, в которой будут храниться результаты рекомендаций по каждому из направлений за все проведённые профориентации, это позволит быстро просматривать и анализировать данные о рекомендациях к поступлению, также разрабатываемый цифровой сервис должен автоматизировать процесс профориентации школьников, что позволит:

- выдавать рекомендации каждому участнику профориентации, не задействуя человеческие ресурсы для проверки результатов;
- упростить процесс проведения профориентации.

Таким образом, определим минимальный перечень требований к веб-приложению:

- анкетирование участника профориентации;
- задача, позволяющая выявить потенциал в направлении обучения «Программная инженерия»;
- задача, позволяющая выявить потенциал в направлении обучения «Информационные системы»;
- задача, позволяющая выявить потенциал в направлении обучения «Информационная безопасность».
- анализ работы участника и выдача рекомендации к поступлению на одно из направлений или выбор другого;
- сохранения результатов рекомендаций по каждому из направлений.

В результате исследования процессов проведения профориентации школьников были проанализированы научные работы в этой области, выделены все недочёты при использовании какой-либо формы профориентационных работ, предложены пути решения и определён минимальный перечень требований к цифровому сервису.

## 1.2 Постановка задачи

Цель проведения профориентации – оказание помощи в выборе вариантов профессионального образования.

Разрабатываемый цифровой сервис должен автоматизировать процессы профориентации школьников, что позволит:

- выдавать рекомендации каждому участнику профориентации, не задействуя человеческие ресурсы для проверки результатов;
- упростить процесс проведения профориентации;
- просматривать статистику по выданным рекомендациям.

Исходя из цели работы, заказчиком установлены и сформулированы следующие задачи:

- провести анализ предметной области;
- оценить аналоги цифровых решений на рынке;
- смоделировать бизнес-процессы проведения профориентации;

– спроектировать и реализовать цифровой сервис.

Таким образом, учитывая всё выше сказанное, было принято решение разработать цифровой сервис профориентации школьников и абитуриентов (ЦС «ИТ-Профи»), который в формате игры позволит выявлять наиболее предпочтительное направления для поступления участника профориентации. Достоинства цифрового сервиса заключаются в том, что профориентация сможет проходить в заочной форме без установки на компьютер испытуемого и доступно в любое время. Базой размещения цифрового сервиса может служить сервер вуза.

По результатам первой главы были исследованы цифровые сервисы, их особенности и атрибуты, рассмотрены процессы проведения профориентации, выделена наиболее подходящая форма для реализации в цифровом сервисе, проведён анализ научных работ связанных с профориентационной деятельностью и описаны задачи, которые необходимо решить в процессе разработки цифрового сервиса.

## 2 Проектирование цифрового сервиса «ИТ-Профи»

### 2.1 Технические требования

По изложенным требованиям и предварительным анализом предметной области к программе были описаны требования в соответствии с государственным стандартом 34.602-89:

Общие сведения:

Полное наименование системы и её условное обозначение:

Полное наименование: цифровой сервис профориентации школьников и абитуриентов.

Условное обозначение: ЦС «ИТ-Профи».

Шифр темы или шифр (номер) договора:

Выпускная квалифицированная работа (ВКР).

Наименование предприятий (объединений) разработчика и заказчика (пользователя) система и их реквизиты:

Федеральное государственное бюджетное образовательное учреждение высшего образования «Иркутский государственный университет путей сообщения» (ФГБОУ ВО ИрГУПС).

Перечень документов, на основании которых создаётся система, кем и когда утверждены эти документы:

Приказ № 4019-с «О закреплении в 2022/2023 учебном году тем ВКР».

Плановые сроки начала и окончания работы по созданию системы:

Начало работ по созданию ПО - 20.12.2022;

Окончание работ - 19.06.2023.

Сведения об источниках порядке финансирования работ:

Финансирование работы не предусмотрено.

Назначение и цели создания системы:

Назначение системы:

Программное обеспечение необходимо для автоматизации процесса профориентации школьников по трём направлениям подготовки: «Программная инженерия», «Информационные системы», «Информационная безопасность».

Цели создания системы:

Целью создания цифрового сервиса является формирование рекомендаций к поступлению и ведение отчётности по количеству выданных рекомендаций на каждое направление за какой-либо период.

Характеристика объектов автоматизации:

краткие сведения об объекте автоматизации или ссылка на документы, содержащие такую информацию:

Объектом автоматизации являются процессы проведения профориентации школьников и абитуриентов по трём направлениям подготовки: «Программная

инженерия», «Информационные системы», «Информационная безопасность», а также формирования отчетов по выданным рекомендациям.

Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды:

условия эксплуатации и окружающая среда регулируются соответствующим СанПиН и трудовым кодексом РФ.

Требования к системе в целом:

Требования к структуре и функционированию системы:

Цифровой сервис «ИТ-Профи» – это новая система, которая позволит автоматизировать процесс проведения профориентации школьников с использованием игровой методики.

Основные функции:

- проведение профориентации;
- формирование рекомендаций.

Требования к программному интерфейсу:

интерфейс должен быть графическим, реализованным в виде GUI для доступа к функциям программы со стороны пользователя и обладать следующими свойствами:

- доступность (простота);
- дружелюбность;
- минимализм (кошелек Миллера);
- отзывчивость (обратная связь);
- естественность.

Требования по взаимодействию и интеграции с другими системами:

Система полностью автономна.

Требования безопасности:

ПО должно иметь защиту от некорректных действий пользователей и ошибочных исходных данных.

Требования к эргономике и технической эстетике:

дизайн приложения ПО должен удовлетворять следующим требованиям по эргономике и технической эстетике:

- быть достаточно «легким» по объему графических элементов и обеспечивать быструю работу приложения;
- обеспечивать минимум усилий пользователя для навигации по приложению;
- должен обладать яркими и привлекательными элементами для концентрации внимания пользователя на каких-то важных моментах, это является особенно важно, так как пользователями будут дети от 12 до 18 лет;
- обладать системой подсказок в местах, где у пользователя потенциально могут возникнуть затруднения.

Дополнительные требования.

Предпочтительной является ООП технология. ПО должно быть создано на модульной основе, позволяющей легко добавлять, изменять функциональные возможности системы.

Требования к функциям (задачам), выполняемым системой:

Языковая поддержка:

система должна содержать материалы на русском языке.

Требования пользователей к системе:

требования к системе, должны быть реализованы:

- функция обработки анкеты;
- функция проверки задания по направлению «Программная инженерия»;
- функция проверки задания по направлению «Информационные системы»;

- функция проверки задания по направлению «Информационная безопасность»;
- функция графического представления данных по рекомендациям на каждое направление за какую-либо дату.

Требования к графическому интерфейсу:

Интерфейс должен быть графическим и обладать следующими свойствами:

- доступность;
- минимализм;
- отзывчивость.

Эффективность. Требования к видам обеспечения:

Программное обеспечение:

ОС Windows, браузер с доступом к интернету

Техническое обеспечение:

50 Мб свободного места на твердотельном накопителе.

Состав и содержание работ по созданию системы:

проектирование – составление теоретической модели ПО;

кодирование – написание ПО.

Порядок контроля и приемки системы:

контроль и приемку осуществляет кафедра информационных систем и защиты информации по проведенным тестированиям.

Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие:

программное средство нуждается в размещении на сервере для работы в сети интернет.

Таким образом, сформулированы и изложены основные требования к цифровому сервису.

## 2.2 Определение опорных точек зрения. Метод VORD

При анализе и формировании требований к цифровому сервису выявлены опорные точки зрения и сервисы, на рисунке 2.1 точки зрения представлены в виде белых круговых областей, а сервисы показаны в виде серых областей.

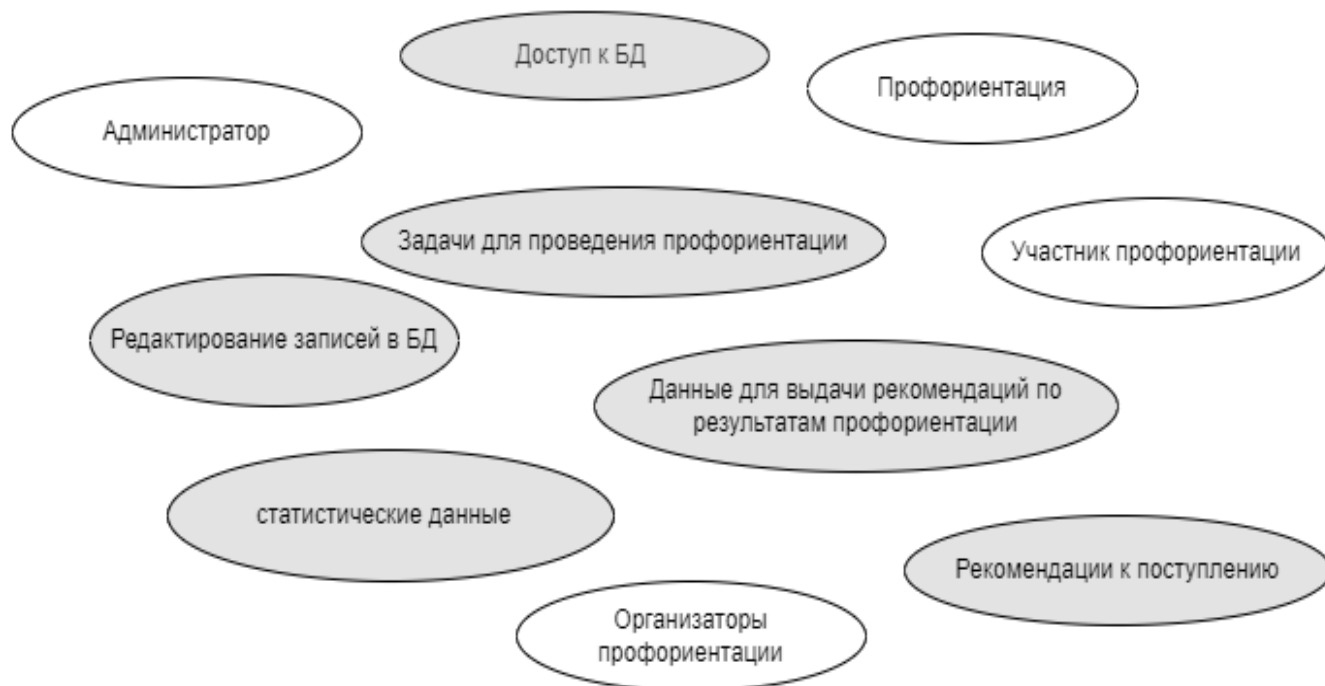


Рисунок 2.1 – Диаграмма идентификации точек зрения

Далее на основе диаграммы идентификации опорных точек зрения, рисунок 2.1, были соотнесены сервисы с определёнными точками зрения, результат представлен на рисунке 2.2.



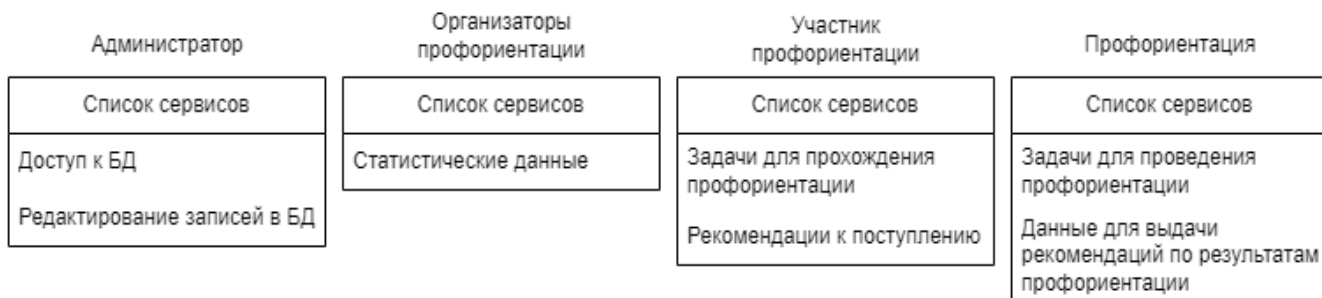


Рисунок 2.2 – Сервисы, соотнесённые с точками зрения

Полученная из опорных точек зрения информация использована для организации точек зрения в иерархию наследования. Это дало возможность проследить общие точки зрения и повторно использовать информацию в иерархии наследования. Сервисы, данные и управляющая информация наследуются подмножеством точек зрения. На рисунке 2.3 показана иерархия точек зрения.



Рисунок 2.3 – Иерархия точек зрения

На основе полученных данных из выявленных опорных точек зрения сформированы основные функциональные требования к приложению. Функциональные требования представлены на диаграмме вариантов использования на рисунке 2.4.

На диаграммах вариантов использования или Use Case отображается взаимодействие между вариантами использования (прецедентами), представляющими функции системы, и действующими лицами (акторами), представляющими людей или системы, получающие или передающие информацию в данную систему. Диаграмма состоит из прецедентов, акторов, интерфейсов и отношений между ними.

Актор – любая внешняя по отношению к моделируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для решения частных задач. Стандартным графическим обозначением актора на диаграммах является фигура проволочного человечка, под которой записывается конкретное имя актера.

Вариант использования - это спецификация функций (сервисов), которые система предоставляет актёру. Обозначается на диаграмме эллипсом, внутри которого содержится его описание, обозначающее выполнение какой-то операции.

Интерфейс служит для указания параметров модели, которые видимы извне без указания их внутренней структуры. В диаграмме вариантов использования интерфейс определяет набор операций, которые предоставляют необходимый набор сервисов или функциональности для акторов. Интерфейсы не могут содержать атрибуты, состояния или направленные ассоциации. Они содержат только операции, которые не определяют детали реализации. Если интерфейс реализует все необходимые операции, то он соединяется с прецедентом сплошной линией. Если же прецедент определяет только тот сервис, который необходим для реализации данного интерфейса, используется пунктирная стрелка.

Примечание может носить самую общую информацию, относящуюся к общему контексту системы, в диаграмме отображается как прямоугольник с подогнутым верхним правым углом.

Отношения отображают на диаграмме связи между актёрами и вариантами использования и бывают четырёх видов:

- ассоциации;
- обобщения;
- включения;
- расширения.

В диаграмме, представленной на рисунке 2.4, присутствуют только три вида отношений - ассоциаций, включения, расширения.

Отношение ассоциации служит для обозначения взаимодействия актёра с вариантом использования.

Отношение включения («include») указывает, что некоторое заданное поведение одного варианта использования обязательно включается в качестве составного компонента в последовательность поведения другого варианта использования.

Отношение расширения («extend») определяет потенциальную возможность включения поведения одного варианта использования в состав другого, то есть дочерний вариант использования может как вызываться, так и не вызываться родительским [10-13].

На диаграмме вариантов использования, рисунок 2.4, выделены четыре актёра:

- участник профориентации – человек, который будет проходить профориентации в разрабатываемом программном продукте, и в результате прохождения получит рекомендации к поступлению на одно из направлений;
- приёмная комиссия – может запросить статистику результатов поступающих;
- специалист по профориентации – может запросить статистику результатов поступающих;
- администратор веб-приложения – имеет доступ ко всем подсистемам.

Основные прецеденты и их описания представлены в таблице 2.1.

Таблица 2.1 – Описание основных прецедентов диаграммы вариантов использования

Прецедент	Актор	Описание
Заполнить форму	Участник профориентации	Перед прохождением профориентации абитуриенту необходимо заполнить анкету
Пройти профориентацию	Участник профориентации	Абитуриенту необходимо по указанным требованиям пройти профориентацию
Прецедент	Актор	Описание
Получить рекомендации	Участник профориентации	Включается в прохождение профориентации, так как по результату необходимо выдавать рекомендации к поступлению
Запросить статистические данные	Организатор профориентации Специалист по профориентации	Данные о результатах профориентации в течение какого-то определённого периода
Указать период для вывода статистических данных	Организатор профориентации Специалист по профориентации	Включается в запрос статистических данных, позволяет просматривать статистику за конкретный период

Также были выделены следующие элементы цифрового сервиса:

- форма для абитуриента (интерфейс), в ней заполняются имя, электронная почта, куда планирует поступать абитуриент и отвечает на некоторый перечень вопросов;
- статистические данные;
- подсистема статистических данных;
- подсистема профориентации.

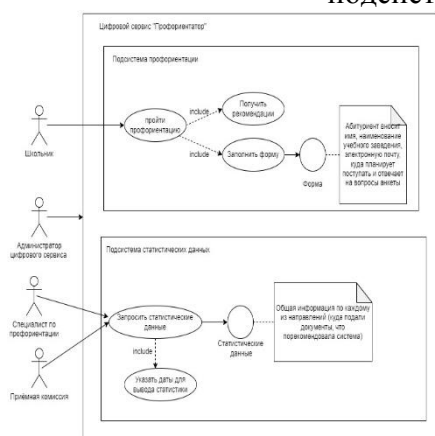


Рисунок 2.4 – Диаграмма вариантов использования

Таким образом, в результате работы выявлены основные точки зрения, которые отображены на диаграмме вариантов использования, представленные на рисунке 2.4.

### 2.3 Моделирование бизнес-процессов проведения профориентации школьников

Функциональная модель системы – это структурированное представление функций в смоделированной системе или предметной области.

Для разработки функциональной модели используется методология IDEF0.

В IDEF0 система представляется как совокупность взаимодействующих работ или функций. Функции анализируются независимо от объектов, которыми они оперируют. Это позволяет четко моделировать логику и взаимодействие процессов организации.

Модель в нотации IDEF0 представляет собой совокупность иерархически упорядоченных и взаимосвязанных диаграмм. Вершина этой древовидной структуры, представляющая собой самое общее описание системы и её взаимодействия с внешней средой, называется контекстной диаграммой. После описания системы в целом проводится разбиение ее на крупные фрагменты. Этот процесс называется функциональной декомпозицией, а диаграммы, которые описывают каждый фрагмент и взаимодействие фрагментов, называются диаграммами декомпозиции. После декомпозиции контекстной диаграммы проводится декомпозиция каждого большого фрагмента системы на более мелкие и так далее до достижения нужного уровня подробности описания. [14]

В работе составлена контекстная диаграмма, которая представлена на рисунке 2.5.

В ней основной функцией является «проводить профориентацию абитуриентов/школьников».

Вход - информация о школьниках, набор направлений для поступления.

Управление - методика профориентации, алгоритмы анализа данных.

Механизмы - информационная система, школьники/абитуриенты, приёмная комиссия.

Выход - статистика для приёмной комиссии.



Рисунок 2.5 – Контекстная диаграмма бизнес-процесса  
 Пояснения к контекстной диаграмме представлены в таблице 2.

Таблица 2.2 – Пояснение контекстной диаграммы

Функция	Описание
Информация о школьнике/абитуриенте	Участником профориентации вносятся имя и направление, на которое он хотел бы поступить
Набор направлений для поступления	Перечень направлений, из которых формируется рекомендация для поступления («Программная инженерия», «Информационные системы», «Информационная безопасность»)
Школьники, приёмная комиссия	Пользователи цифрового сервиса
Функция	Описание
Информационная система	Цифровой сервис профориентации школьников и абитуриентов
Статистика для приёмной комиссии	Количество абитуриентов, получивших рекомендации по каждому из направлений

После описания основной функции выполняется функциональная декомпозиция, то есть определяются функции, из которых состоит основная. Диаграмма декомпозиции изображена на рисунке 2.6.

Процесс «проводить профориентацию школьников/абитуриентов» состоит из трёх этапов.

Первый этап – собрать данные. Школьники заполняют поля необходимыми данными.

Второй этап – провести профориентацию и выдать рекомендации, в нём участвует информационная система и школьник/абитуриент. Профориентация проходит в формате игры, она делится на три части. Первая – задача, в которой участнику профориентации необходимо показать свои умения в области алгоритмизации, для выдачи рекомендации по направлению «Программная инженерия». Вторая часть – задача, направленная на выявление умений определять основных пользователей системы и их функции, для выдачи рекомендации по направлению «Информационные системы». Третья часть – задача, которая позволит выявить знания в сфере безопасности информации, для выдачи рекомендации по направлению «Информационная безопасность». После выполнения всех задач система анализирует ответу участника профориентации и формирует рекомендацию, которая сохраняется для статистики в базе данных системы.

Третий этап – проанализировать результаты и выдать результаты, в котором участвует информационная система, она получает от приёмной комиссии запрос на формирования данных за какой-то конкретный период, либо за всё время, анализирует результаты испытуемых и выдает количество рекомендаций по каждому из направлений. Результат этого этапа - статистика для приёмной комиссии.

Далее выполнена декомпозиция диаграммы, рисунок 2.6. Результат декомпозиции представлен на рисунке 2.7, декомпозиции подлежал блок «Проведения профориентации», в нём было выделено три функции:

- заполнение анкеты – участник отвечает на несколько общих вопросов, ответы в дальнейшем оцениваются и влияют на конечный результат профориентации;
- прохождение заданий – задания представляют собой решение трёх задач, для оценки по каждому направлению;
- получение рекомендаций – по результатам профориентации выдается рекомендация к поступлению на одно из трёх направлений подготовки «Программная инженерия», «Безопасность информации», «Информационные системы».

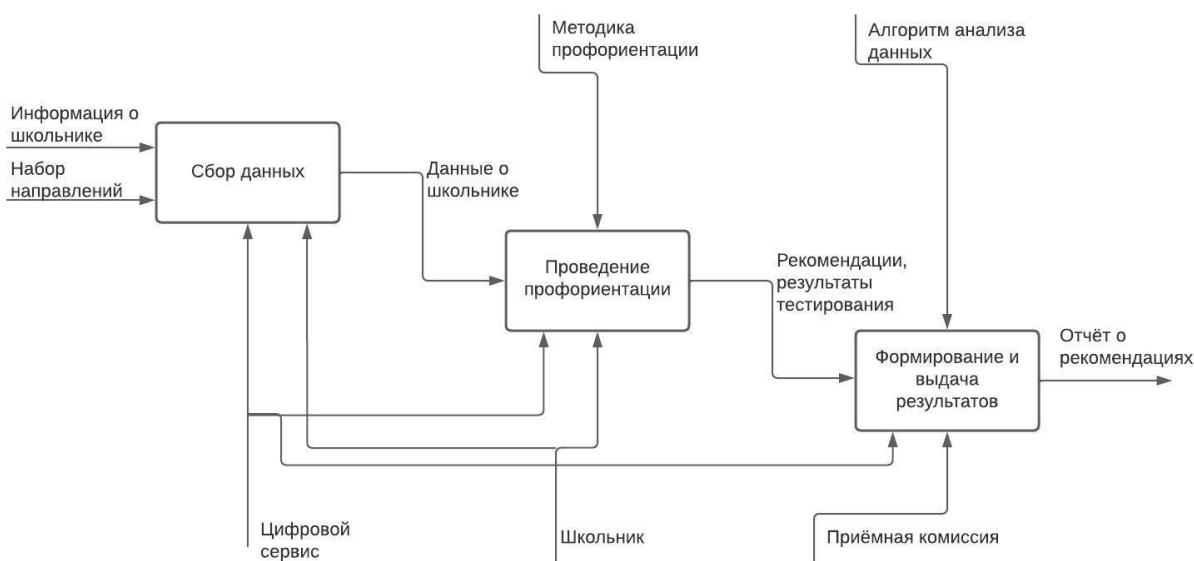


Рисунок 2.6 – Диаграмма декомпозиции 1-ого уровня

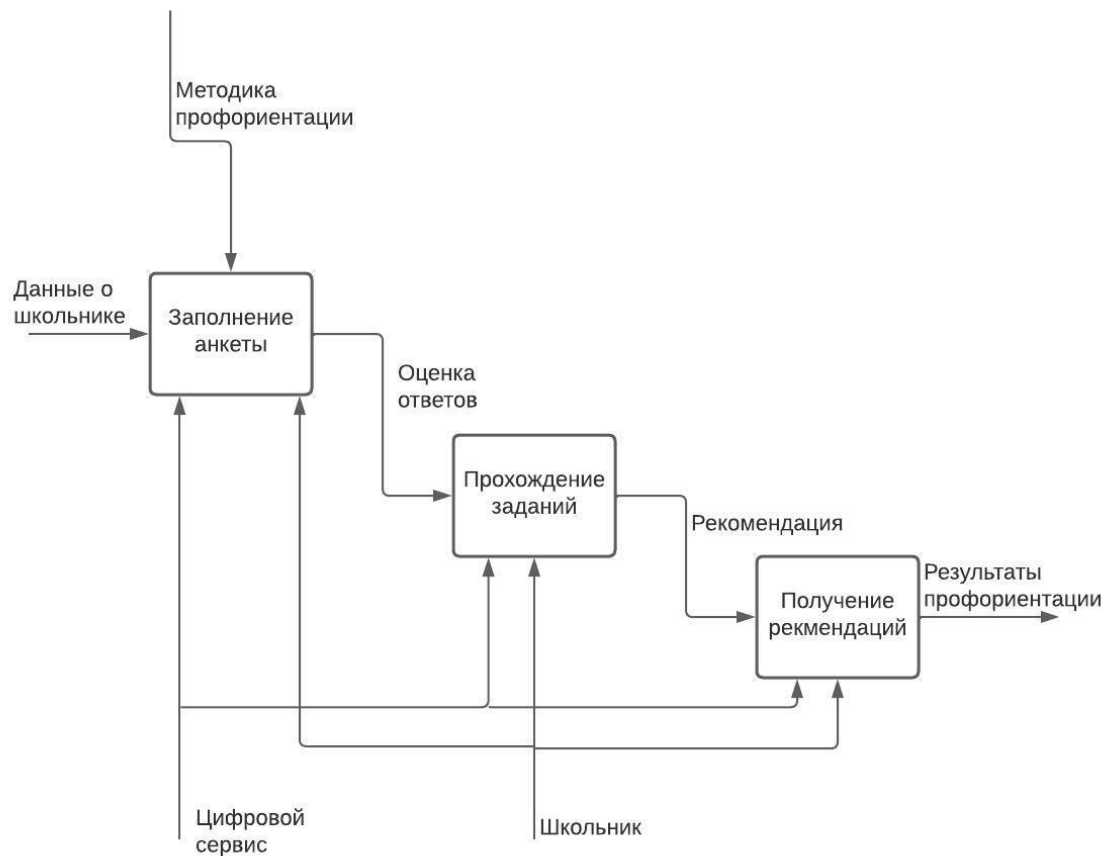


Рисунок 2.7 – Диаграмма декомпозиции 2-ого уровня

Далее была разработана блок-схема для отображения алгоритма работы разрабатываемого цифрового сервиса, на ней наглядно показаны переходы от одного процесса к другому. После перехода к главной странице идёт разделение процессов на две ветки, левая – это проведение профориентации, правая – просмотр отчётов о выдачи рекомендаций. Блок-схема представлена на рисунке 2.8.

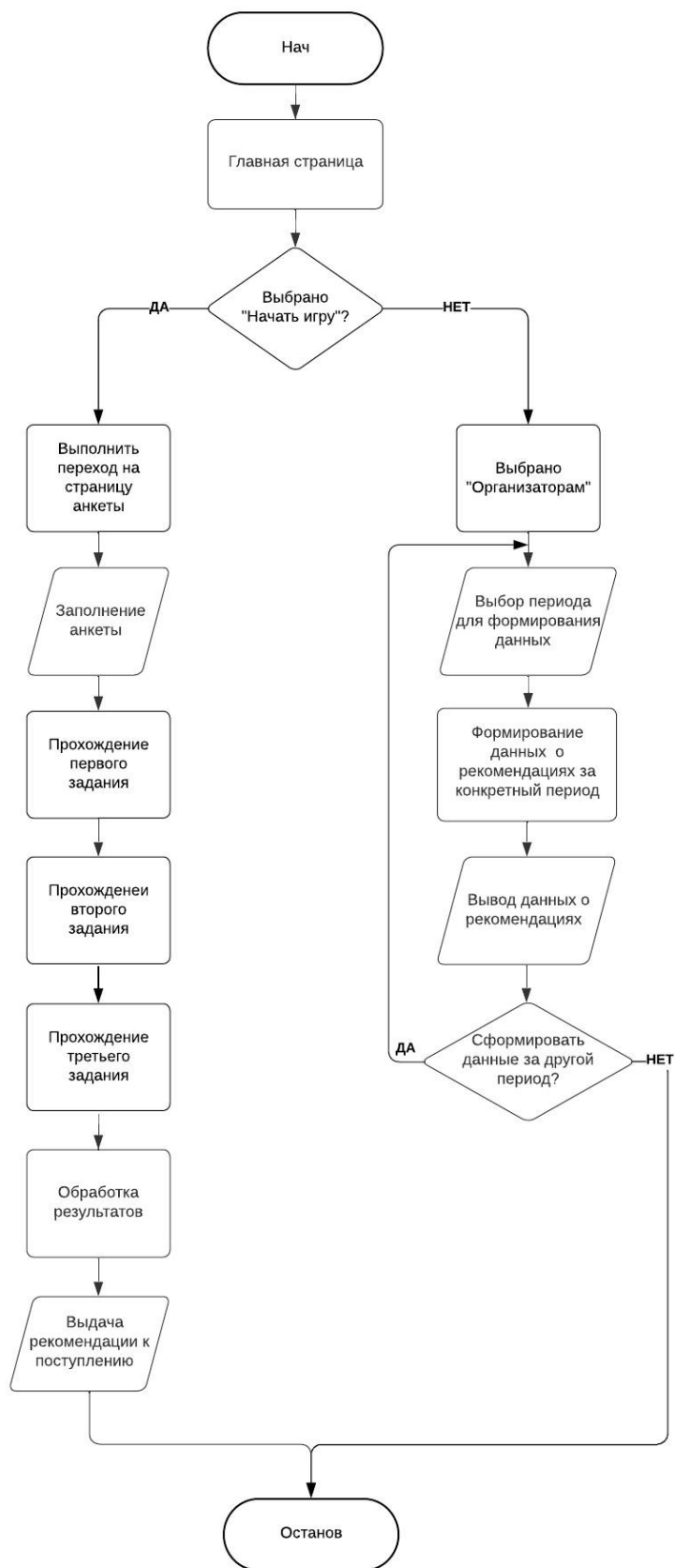


Рисунок 2.8 – Блок-схема алгоритма работы цифрового сервиса профориентации школьников и абитуриентов

Таким образом, проведено моделирование бизнес-процессов и их декомпозиция, а также спроектирован базовый алгоритм работы цифрового сервиса «ИТ-Профи».

## 2.4 Выбор средств разработки

Языков программирования, используемых для серверной веб-разработки, достаточно много: PHP, Java, Python и другие.

В техническом плане для большинства проектов нет каких-либо ограничений при выборе языка, то есть практически любая функциональность сайта или приложения может быть успешно реализована на любом из них, поэтому выбор языка не накладывает никаких лимитаций на проект.

PHP – скриптовый язык общего назначения, часто применяемый для разработки веб-сайтов. PHP – интерпретируемый язык программирования, позволяющий создавать программы в процедурном и объектно-ориентированном стиле.

Его главными преимуществами перед подобными языками являются простота в изучении и довольно высокая производительность. Помимо этого, можно отметить следующие плюсы данного языка:

- мощный и гибкий – как мы уже писали, этот язык способен обслужить небольшой личный блог, но при этом же спокойно себя чувствует в больших ресурсах: интернет-магазинах, соцсетях, порталах и т. д.;

- свободный – данный язык является полностью бесплатным и распространяется со свободной лицензией, поэтому его смело могут применять как частные лица, так и коммерческие организации;

- простой – чтобы уловить логику его работы и понимать основной синтаксис, хватит 2-3 недель обучения; конечно, чтобы понять его «глубже», потребуется больше времени;

- дополняемость – PHP имеет большое количество дополнительных расширений и библиотек, которые увеличивают его функциональность;

- не требователен – этот язык можно применять на всех известных видах серверов;

- простота редактирования – писать и редактировать этот код можно в любом редакторе текста.

Однако язык PHP неидеальный, и это подтверждают его следующие минусы:

- на нем невозможно создать десктопное приложение или какой-нибудь системный компонент;

- у приложений на PHP более низкая защищенность, чем у приложений, написанных с использованием других языков;

- слабая возможность работы с глобальными исключениями [15].

Python – высокоуровневый интерпретируемый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода.

Выделяются следующие преимущества языка Python:

- простота синтаксиса – код языка чистый и понятный, без лишних символов и выражений;

- расширяемость и гибкость – Python можно легко расширить для взаимодействия с другими программными системами или встроить в программы в качестве компонента;

- интерпретируемость и кроссплатформенность – интерпретатор Python есть для всех популярных платформ и по умолчанию входит в большинство дистрибутивов Linux;

- стандартизованность – у Python есть единый стандарт для написания кода (Python Enhancement Proposal или PEP), благодаря чему язык остаётся читабельным даже при переходе от одного программиста к другому;

- open source – у интерпретатора Python открытый код, то есть любой, кто заинтересован в развитии языка, может поучаствовать в его разработке и улучшении;

- широта применения – наиболее широко Python используется в web-разработке, работе с данными, автоматизации бизнес-процессов и геймдеве.

- востребованность на рынке труда и поддержка гигантами IT-сферы – Python-разработчики востребованы во многих проектах и им несложно найти работу.



#### Недостатки языка:

- низкая производительность – Python требует высоких вычислительных мощностей серверов и компьютеров;
- динамическая типизация – Python относится к языкам с динамической типизацией, что делает его невероятно гибким при разработке, но при этом он потребляет много ресурсов и имеет низкую скорость выполнения программы;
- глобальная блокировка интерпретатора (GIL) – это способ синхронизации потоков, который используется в некоторых интерпретируемых языках программирования, например, в Python и Ruby, Хотя GIL является самым простым способом избежать конфликтов при одновременном обращении разных потоков к одним и тем же участкам памяти, у такого подхода есть недостаток – ограничение параллельности вычислений [16].

Java – широко используемый язык программирования для написания интернет-приложений. Язык Java широко использовался на протяжении более двух десятилетий. Миллионы приложений Java используются и сегодня. Java – это кроссплатформенный, объектно-ориентированный и сетевый язык, который сам по себе может использоваться как платформа. Это быстрый, безопасный и надежный язык программирования для всего: от мобильных приложений и корпоративного ПО до приложений для работы с большими данными и серверных технологий.

#### Плюсы Java:

- концепция ООП (объектно-ориентированного программирования) – это значит, что программист сам определяет тип данных, его структуру и набор применяемых к нему функций. Это позволяет многократно использовать одни и те же объекты в разных программах, более удобно для организации структуры программ (особенно с большим объемом кода), помогает избегать ошибок и упрощает поддержку и модернизацию старого кода;
- это язык высокого уровня, т.е., он больше похож на человеческую речь, а не на машинный код – следовательно, у него сравнительно простой синтаксис, что делает его быстрым для освоения и удобным для написания кода, его чтения и обслуживания;
- безопасность – у Java есть несколько функций, которые ликвидируют часто встречающиеся уязвимости;
- удобство для распределённого программирования – этот язык изначально создавался для совместной работы (в том числе удалённой), поэтому он позволяет совместно использовать данные и программы несколькими компьютерами одновременно;
- принцип «написать один раз и использовать везде» – написанное на Java приложение можно запустить на любой поддерживающей его платформе;
- стабильное и постоянно развивающееся сообщество – по многочисленности и активности с ним мало кто может соперничать.

#### Минусы Java:

- низкая скорость – все высокоуровневые языки приходится компилировать с помощью виртуальной машины, что плохо сказывается на их производительности. Java – не исключение, кроме того, у него есть и некоторые собственные особенности, вызывающие дополнительные проблемы с производительностью;
- многословие (verbosity) – сходство с естественными языками делает Java проще для изучения и понимания, но также ведёт и к тому, что он содержит много лишней информации и довольно громоздок;
- платность для коммерческого использования (с 2019 года) [17].

На этапе анализа предметной области было выявлено, что у приложения должна быть своя база данных. База данных – это упорядоченная совокупность информации, хранящейся в виде множеств, каждое из которых содержит записи унифицированного вида. Системы управления базами данных (СУБД) предоставляют программисту инструментальный для

создания, обновления и обработки больших объемов информации, имеющей сложную структуру.

В классической теории выделяют три типа, три структуры баз данных: иерархическую, сетевую и реляционную. В настоящее время доминирующее положение занимают реляционные базы данных [18].

Реляционная база данных – это набор данных с предопределенными связями между ними. Эти данные организованы в виде набора таблиц, состоящих из столбцов и строк. В таблицах хранится информация об объектах, представленных в базе данных. В каждом столбце таблицы хранится определенный тип данных, в каждой ячейке – значение атрибута. Каждая строка таблицы представляет собой набор связанных значений, относящихся к одному объекту или сущности. Каждая строка в таблице может быть помечена уникальным идентификатором, называемым первичным ключом, а строки из нескольких таблиц могут быть связаны с помощью внешних ключей. К этим данным можно получить доступ многими способами, и при этом реорганизовывать таблицы БД не требуется [11].

В настоящее время самыми распространенными реляционными базами данных являются MySQL, Oracle и PostgreSQL.

MySQL является наиболее приспособленной для применения в среде web СУБД, к её преимуществам относят следующие:

- многопоточность, поддержка нескольких одновременных запросов;
- оптимизация связей с присоединением многих данных за один проход;
- записи фиксированной и переменной длины;
- ODBC драйвер;
- гибкая система привилегий и паролей;
- гибкая поддержка форматов чисел, строк переменной длины и меток времени;
- интерфейс с языками C и Perl, PHP;
- быстрая работа, масштабируемость;
- совместимость с ANSI SQL;
- бесплатная в большинстве случаев;
- хорошая поддержка со стороны провайдеров услуг хостинга;
- быстрая поддержка транзакций через механизм InnoDB[12].

Oracle – это объектно-реляционная СУБД, созданная компанией Oracle. Данная СУБД имеет ряд преимуществ.

Oracle поддерживает самые большие базы данных. СУБД способна поддерживать любых пользователей, в любом количестве, которые при этом одновременно выполняют разные задачи. В Oracle не происходит соперничества между разными видами данных.

СУБД Oracle хорошо обрабатывает транзакции. Система сохраняет высокую производительность, в результате чего пользователи не страдают от низкой скорости обработки.

Система обладает высокой степенью готовности. В разных установках, продолжительность работы Oracle индивидуальная. Так, например, в некоторых, система способна работать круглосуточно. При этом откат БД или какие-либо сбои системы не приводят к остановке работы базы.

Эта система обладает локальной управляемостью. Например, чтобы перезагрузить данные какого-то определенного приложения, не обязательно отключать всю систему. Администратору достаточно выключить доступ к нужному приложению и выполнить с ним требуемые манипуляции.

Данная СУБД легко переносится с одной ОС на другую. Приложения, которые были разработаны специально для Oracle, легко переносятся на любую операционную систему с минимальными изменениями, а иногда даже без них[13].

PostgreSQL – это популярная свободная объектно-реляционная система управления базами данных, к её преимуществам относят следующие:

- поддержка БД неограниченного размера;
- мощные и надёжные механизмы транзакций и репликации;

–расширяемая система восторенных языков программирования и поддержка загрузки С-совместимости модулей;

–лёгкая расширяемость [14].

Язык программирования выбран по требованиям заказчика, выбор СУБД, обусловлен политикой импортозамещения в сложившейся политической обстановке, выбор данного СУБД гарантирует, что не возникнет никаких проблем с эксплуатацией программного обеспечения.

По результатам второй главы сформированы основные требования цифровому сервису, смоделированы бизнес-процессы профориентационных работ со школьниками, спроектирован алгоритм работы цифрового сервиса, выбран инструментарий для разработки.

Образец типовых вопросов для защиты курсовых работ

1. Определение и описание основных этапов разработки программы.
2. Системные и технические требования для работы программы.
3. Выбор и обоснование моделей и их иерархии.
4. Разработка и обоснование алгоритмов и их иерархии.
5. Описание моделей данных, их атрибутов, иерархии.
6. Разработка и обоснование общей схемы алгоритма программы.
7. Дизайн пользовательского интерфейса программы.
8. Выбор и обоснование стратегии тестирования программы, ее реализация.
9. Оценка направлений усовершенствования программы.

### **3.5 Перечень теоретических вопросов к зачету**

(для оценки знаний)

Раздел 1 «Введение в дисциплину»

1. Жизненный цикл ПО
2. Основные этапы разработки программного обеспечения
3. Системное проектирование приложений
4. Основные оценочные требования к ПО
5. Основные спецификации ПО

Раздел 2 Стандарты разработки, технико-экономическое обоснование и требования к ПО. Планирование жизненного цикла. Проектирование ПО

6. Международные стандарты разработки ПО
7. Отечественные стандарты разработки ПО
8. Взаимосвязь международных и отечественных стандартов
9. Профили стандартов
10. Обоснование разработки ПО
11. Основные показатели экономической эффективности программных средств
12. Каскадная модель разработки ПО
13. Спиральная модель разработки ПО
14. Модель эволюционного прототипирования
15. Инкрементная модель разработки ПО
16. Модель RAD
17. Особенности моделей жизненного цикла
18. Язык UML и его применение в программной инженерии
19. Диаграммы ПО
20. Применение языка UML в системном проектировании ПО
21. Применение MS Project в системном проектировании ПО

### **3.6 Перечень типовых простых практических заданий к зачету**

(для оценки умений)

1. Общая теория и примеры технологий разработки ПО (Microsoft Solution Framework
2. (MSF),

3. Rational Unified Process (RUP), SADT (IDEF<sub>x</sub>), другие)
4. Диаграммы прецедентов, состояний, активности, взаимодействия, классов, компонент, последовательности действий и другие на языке UML.
5. Назначение профилей стандартов жизненного цикла в программной инженерии.
6. Общая теория о профилях стандартов и их назначении.
7. Жизненный цикл профилей стандартов систем и программных средств.
8. Модель профиля стандартов жизненного цикла сложных программных средств
9. Ресурсы жизненного цикла программного средства.
10. Основные ресурсы ПО.
11. Основные статьи затрат ПО. Ресурсы специалистов в жизненном цикле ПО.
12. Системное планирование и управление ПО.
13. Планирование проекта.
14. Процесс системного проектирования ПО.
15. Управление проектом, выбор руководителя.
16. Описание процессов управления качеством.

### **3.7 Перечень типовых практических заданий к зачету**

(для оценки навыков и (или) опыта деятельности)

1. Требования, предъявляемые к информационным системам: гибкость, надежность, эффективность, безопасность.
2. Понятие архитектуры информационной системы. Способы представления. Примеры.
3. Понятие жизненного цикла информационных систем. Понятие проекта. Классификация проектов.
4. Процессы жизненного цикла информационных систем. Основные процессы жизненного цикла.
5. Процессы жизненного цикла информационных систем. Вспомогательные процессы жизненного цикла.
6. Процессы жизненного цикла информационных систем. Организационные процессы жизненного цикла.
7. Структура жизненного цикла информационных систем. Начальная стадия.
8. Структура жизненного цикла информационных систем. Стадия уточнения.
9. Структура жизненного цикла информационных систем. Стадия конструирования.
10. Структура жизненного цикла информационных систем. Стадия ввода в эксплуатацию.
11. Методология быстрой разработки информационных систем. Фазы жизненного цикла информационных систем в рамках методологии. Фаза проектирования.
12. Методология быстрой разработки информационных систем. Фазы жизненного цикла информационных систем в рамках методологии. Фаза построения.
13. Методология быстрой разработки информационных систем. Фазы жизненного цикла информационных систем в рамках методологии. Фаза внедрения.
14. CASE-технологии проектирования информационных систем. Характеристика CASE-средств. Примеры.

### **3.8 Перечень теоретических вопросов к экзамену**

(для оценки знаний)

Раздел 3 «Проблемы разработки. Тестирование и документирование ПО»

1. Классификация типов программного обеспечения
2. Цели и задачи технологий разработки ПО. Особенности современных крупных проектов ИС
3. Основные оценочные требования к ПО
4. Основные спецификации ПО
5. Основные показатели экономической эффективности программных средств
6. Рациональный Унифицированный Процесс. Динамические аспекты процессов: структура ЖЦ, стадии, итерации и контрольные точки.
7. Развитие инкрементального подхода. XP-процессы
8. Проектирование ИС на основе объектно-ориентированного подхода. Сопоставление и взаимосвязь структурного и объектно-ориентированного подходов
9. Проектирование ИС на основе объектно-ориентированного подхода. Объектно-ориентированная разработка программ. Объектно-ориентированные языки программирования. Объектно-ориентированные методологии разработки программных систем. CASE - средства разработки ПО
10. Рациональный Унифицированный Процесс. Динамические аспекты процессов: структура ЖЦ, стадии, итерации и контрольные точки
11. Рациональный Унифицированный Процесс. Статическое содержание процесса: виды деятельности (технологические операции), рабочие продукты, исполнители и дисциплины (технологические процессы).
12. Ресурсы ПО
13. Проект. Состав и структура коллектива разработчиков, их функции
14. Управление персоналом (Staff Resource Management), планирование организационной структуры.
15. Человеческий фактор в управлении проектами. Задача n-личностей. Закон Брукса. Подходы к управлению группами и руководству ими
16. Выполнение оценки проекта на основе LOC- и FP-метрик
17. Отладка ПО
18. Оценка качества ПО
19. Риски в разработке ПО
20. Корректность программных средств. Первичные и вторичные ошибки
21. Качество программного продукта. Критерии качества ПО
22. Методы оценки качества программного обеспечения.
23. Сертификация фирм разработчиков по модели качества СММ
24. Измерения, меры и метрики. Размерно-ориентированные метрики. Функционально-ориентированные метрики
25. Ошибки при разработке ПО
26. Виды ошибок
27. Верификация программных кодов
28. Методики тестирования программного обеспечения
29. Задачи мониторинга и сопровождения ПО
30. Состав документов по ПО
31. Содержание документов по ПО
32. Документация, создаваемая в процессе разработки программных средств. Документы управления разработкой ПС. Документы, входящие в состав ПС
33. Пользовательская документация. Помощь в освоении ПО
34. Документация по сопровождению программных средств. Порядок ее разработки.

### **3.9 Перечень типовых простых практических заданий к экзамену** (для оценки умений)

- 1** Объясните, что значит технологичность ПО
- 2** Объясните, чем отличаются разные типы ПО
- 3** Раскройте эксплуатационные требования к ПО
- 4** Раскройте назначение и содержание предпроектного обследования
- 5** Перечислите основные элементы диаграммы UML и их назначение

- 6 Перечислите основные отношения диаграммы UML и их назначение
- 7 Укажите виды связности модулей программы
- 8 Разъясните понятие сцепления модуля
- 9 Раскройте понятия абстрактного и корневого классов в UML
- 10 Разъясните понятие сообщения в UML
- 11 Перечислите виды тестирования программы
- 12 Перечислите методики структурного тестирования программы
- 13 Перечислите методики функционального тестирования программы
- 14 Раскройте способы тестирования циклов и условий
- 15 Перечислите метрики количественной оценки программного кода
- 16 Охарактеризуйте подходы к оценке качества ПО

### **3.10 Перечень типовых практических заданий к экзамену** (для оценки навыков и (или) опыта деятельности)

- 1 Предложите пример прямого и обратного проектирования ПО
- 2 Предложите пример документа технико-экономического обоснования проекта сложного программного средства
- 3 Приведите пример спецификации требований к программной системе
- 4 Предложите пример оценки рисков при разработке программного комплекса
- 5 Предложите пример анализа и оценки корректности программы
- 6 Предложите пример оценки качества ПО, обосновав используемую методику
- 7 Предложите структуру базы данных для управления конфигурацией проекта ПО
- 8 Предложите пример контракта с заказчиком на обеспечение жизненного цикла программного комплекса

#### 4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности

В таблице приведены описания процедур проведения контрольно-оценочных мероприятий и процедур оценивания результатов обучения с помощью оценочных средств в соответствии с рабочей программой дисциплины.

Наименование оценочного средства	Описания процедуры проведения контрольно-оценочного мероприятия и процедуры оценивания результатов обучения
Доклад	Защита докладов, предусмотренных рабочей программой дисциплины, проводится во время практических занятий. Преподаватель на практическом занятии, предшествующем занятию проведения контроля, доводит до обучающихся: тему докладов и требования, предъявляемые к их выполнению и защите
Лабораторная работа	Защита лабораторных работ проводится во время лабораторных занятий. Во время проведения защиты лабораторной работы пользоваться учебниками, справочниками, конспектами лекций, тетрадями не разрешено. Преподаватель на лабораторной работе, предшествующей занятию проведения защиты лабораторной работы, доводит до обучающихся: номер защищаемой лабораторной работы, время на защиту лабораторной работы. Преподаватель информирует обучающихся о результатах защиты лабораторной работы сразу после ее контрольно-оценочного мероприятия
Курсовая работа	Ход выполнения разделов курсовой работы в рамках текущего контроля оценивается преподавателем исходя из объемов выполненных работ в соответствии со шкалами оценивания. Преподаватель информирует обучающихся о результатах оценивания выполнения курсового проекта сразу после контрольно-оценочного мероприятия. В ходе защиты курсовой работы обучающийся делает доклад протяженностью 5 – 7 минут. Преподаватель ставит окончательную оценку за курсовую работу после завершения защиты, учитывая уровень ее защиты

Для организации и проведения промежуточной аттестации составляются типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.

Перечень теоретических вопросов и типовые практические задания разного уровня сложности для проведения промежуточной аттестации обучающиеся получают в начале семестра через электронную информационно-образовательную среду ИрГУПС (личный кабинет обучающегося).

#### Описание процедур проведения промежуточной аттестации в форме зачета и оценивания результатов обучения

При проведении промежуточной аттестации в форме зачета преподаватель может воспользоваться результатами текущего контроля успеваемости в течение семестра. С целью использования результатов текущего контроля успеваемости, преподаватель подсчитывает среднюю оценку уровня сформированности компетенций обучающегося (сумма оценок, полученных обучающимся, делится на число оценок).

#### Шкала и критерии оценивания уровня сформированности компетенций в результате изучения дисциплины при проведении промежуточной аттестации в форме зачета по результатам текущего контроля (без дополнительного аттестационного испытания)

Средняя оценка уровня сформированности компетенций по результатам текущего контроля	Шкала оценивания
Оценка не менее 3,0 и нет ни одной неудовлетворительной оценки по текущему контролю	«зачтено»
Оценка менее 3,0 или получена хотя бы одна неудовлетворительная оценка по текущему контролю	«не зачтено»

Если оценка уровня сформированности компетенций обучающегося не соответствует критериям получения зачета без дополнительного аттестационного испытания, то промежуточная аттестация проводится в форме собеседования по перечню теоретических вопросов и типовых практических задач или в форме компьютерного тестирования.

Промежуточная аттестация в форме зачета с проведением аттестационного испытания проходит на последнем занятии по дисциплине.

При проведении промежуточной аттестации в форме компьютерного тестирования вариант тестового задания формируется из фонда тестовых заданий по дисциплине случайным образом, но с условием: 50 % заданий должны быть заданиями открытого типа и 50 % заданий – закрытого типа.

### **Описание процедур проведения промежуточной аттестации в форме экзамена и оценивания результатов обучения**

Промежуточная аттестация в форме экзамена проводится путем устного собеседования по билетам или в форме компьютерного тестирования.

При проведении промежуточной аттестации в форме собеседования билеты составляются таким образом, чтобы каждый из них включал в себя теоретические вопросы и практические задания.

Билет содержит: два теоретических вопроса для оценки знаний. Теоретические вопросы выбираются из перечня вопросов к экзамену; два практических задания: одно из них для оценки умений (выбирается из перечня типовых простых практических заданий к экзамену); другое практическое задание для оценки навыков и (или) опыта деятельности (выбираются из перечня типовых практических заданий к экзамену).


Распределение теоретических вопросов и практических заданий по экзаменационным билетам находится в закрытом для обучающихся доступе. Разработанный комплект билетов (25-30 билетов) не выставляется в электронную информационно-образовательную среду ИрГУПС, а хранится на кафедре-разработчике фондов оценочных средств.

На экзамене обучающийся берет билет, для подготовки ответа на экзаменационный билет обучающемуся отводится время в пределах 45 минут. В процессе ответа обучающегося на вопросы и задания билета, преподаватель может задавать дополнительные вопросы.

Каждый вопрос/задание билета оценивается по четырехбалльной системе, а далее вычисляется среднее арифметическое оценок, полученных за каждый вопрос/задание. Среднее арифметическое оценок округляется до целого по правилам округления.

При проведении промежуточной аттестации в форме компьютерного тестирования вариант тестового задания формируется из фонда тестовых заданий по дисциплине случайным образом, но с условием: 50 % заданий должны быть заданиями открытого типа и 50 % заданий – закрытого типа.

### **Образец экзаменационного билета**

	<p><b>Экзаменационный билет № 1</b> по дисциплине <b>«Методология программной инженерии»</b></p>	<p>Утверждаю: Заведующий кафедрой «Информационные системы и защита информации» ИрГУПС</p>
<p>1. Модели ЖЦ ПО. Инкрементальная модель. Содержание этапов создания ПО 2. ЕСПД 3. Перечислите основные элементы диаграммы UML и их назначение 4. Охарактеризуйте подходы к оценке качества ПО</p>		