

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Попов Анатолий Николаевич  
Должность: директор  
Дата подписания: 16.06.2022 18:11:37  
Уникальный программный ключ:  
1e0c38dcaae73cee1e5e09c1d5873fc7497ba8

Приложение 2  
к рабочей программе дисциплины

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**  
**ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**  
**Технология разработки программного обеспечения**

---

*(наименование дисциплины(модуля))*

Направление подготовки / специальность

**09.03.03 Прикладная информатика**  
*(код и наименование)*

Направленность (профиль)/специализация

**Прикладная информатика на железнодорожном транспорте**  
*(наименование)*

## Содержание

1. Пояснительная записка.
2. Типовые контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций.
3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации.

## 1. Пояснительная записка

Цель промежуточной аттестации – оценивание промежуточных и окончательных результатов обучения по дисциплине, обеспечивающих достижение планируемых результатов освоения образовательной программы.

### Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции
УК-2.1 Решает ситуационные задачи с учетом трудовых и социальных факторов в рамках нормативно-правового регулирования
УК-2.2 Выбирает оптимальные варианты действий в соответствии с предписаниями правовых норм
ПК-1.5 Оформляет техническую документацию

### Результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
<b>УК-2.1</b> Решает ситуационные задачи с учетом трудовых и социальных факторов в рамках нормативно-правового регулирования	<b>УК-2.1.1</b> Обучающийся знает: Методы планирования проектных работ по разработке программного обеспечения. Методы функционального проектирования.	Вопросы тестирования №(1-5)
	<b>УК-2.1.2</b> Обучающийся умеет: Определять круг задач в рамках разработки программного обеспечения, планировать собственную деятельность исходя из имеющихся ресурсов. Соотносить главное и второстепенное, решать поставленные задачи в рамках разработки программного обеспечения.	Задания №(1-2)
	<b>УК-2.1.3</b> Обучающийся владеет: Выявления требований потребителей к системе и их интересов. Определения значимых показателей деятельности объекта автоматизации, на изменение которых направлен проект.	Задания №(3-4)
<b>УК-2.2</b> Выбирает оптимальные варианты действий в соответствии с предписаниями правовых норм	<b>УК-2.2.1.</b> Обучающийся знает: Необходимые для разработки программного обеспечения правовые нормы.	Вопросы тестирования №(6-10)
	<b>УК-2.2.2.</b> Обучающийся умеет: Планировать проектные работы; выбирать методики разработки требований к системе и шаблоны документов требований к системе; формулировать цели, исходя из анализа проблем, потребностей и возможностей.	Задания №(5)
	<b>УК-2.2.3</b> Обучающийся владеет: Создавать графические документы в программах растровых и векторных изображений.	Задания №(6-7)

	Разработки блок-схемы разрабатываемых систем. Реинжиниринга разработанного программного обеспечения для решения технических задач.	
<b>ПК-1.5</b> Оформляет техническую документацию	<b>ПК-1.5.1.</b> Обучающийся знает: Технические требования к пользовательскому интерфейсу. Стандарты регламентирующие требования к эргономике разработки взаимодействия человек-систем. Архитектуру целевой аппаратной платформы, для которой разрабатывается программное обеспечение.	Вопросы тестирования №(11-14)
	<b>ПК-1.5.2.</b> Обучающийся умеет: Разрабатывать графический дизайн интерфейсов. Создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов. Осуществлять отладку программных продуктов для целевой операционной системы.	Задания №(8-9)
	<b>ПК-1.5.3</b> Обучающийся владеет: опытом применения нормативной базы и решения задач в области разработки программного обеспечения. Составления и согласования перечня требований к системе.	Задания №(10)

Промежуточная аттестация (экзамен, зачёт) проводится в одной из следующих форм:

- 1) ответ на билет, состоящий из теоретических вопросов и практических заданий;
- 2) выполнение заданий в ЭИОС СамГУПС.

## 2. Типовые<sup>1</sup> контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций

### 2.1 Типовые вопросы (тестовые задания) для оценки знаниевого образовательного результата

Проверяемый образовательный результат

Код и наименование индикатора достижения компетенции	Образовательный результат
<b>УК-2.1.1</b>	Обучающийся знает: Методы планирования проектных работ по разработке программного обеспечения. Методы функционального проектирования.
<p>Задание 1. С какого момента появляется жизненный цикл?</p> <ol style="list-style-type: none"> <li>1. С момента осознания появления данного ПО;</li> <li>2. С момента его полного выхода из употребления.</li> </ol> <p>Задание 2. Стадии жизненного цикла?</p> <ol style="list-style-type: none"> <li>1.Определения требований и спецификаций; проектирования; программирования и сопровождения;</li> <li>2.Надежность; технологичность; правильность.</li> </ol> <p>Задание 3. Сколько стадий в ЖЦ программного обеспечения?</p>	

<sup>1</sup>Приводятся типовые вопросы и задания. Оценочные средства, предназначенные для проведения аттестационного мероприятия, хранятся на кафедре в достаточном для проведения оценочных процедур количестве вариантов. Оценочные средства подлежат актуализации с учетом развития науки, образования, культуры, экономики, техники, технологий и социальной сферы. Ответственность за нераспространение содержания оценочных средств среди обучающихся университета несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

4;  
5;  
6.  
Задание 4. Что делает программный–аналитик?  
1.анализирует и проектирует комплекс взаимосвязанных программ;  
2.разработчик формальных постановок задач, требующих реализации на ЭВМ.  
Задание 5. Кто является разработчиком формальных постановок задач, требующих реализацию на ЭВМ?  
1. Программист–аналитик;  
2. Инженер;  
3. Постановщик задач.

Код и наименование индикатора достижения компетенции	Образовательный результат
<b>УК-2.2.1</b>	Обучающийся знает:Необходимые для разработки программного обеспечения правовые нормы.

Задание 6. Какие требования устанавливаются на первой стадии?  
1.по надежности;  
2.технологичности;  
3.правильности;  
4.универсальности;  
5.эффективности;  
6. информационной согласованности;  
7. все верно.  
Задание 7. Что определяют функциональные спецификации?  
1.функции, которые должно выполнять ПО, т.е. в них определяется, что надо делать системе, а не то, как это делать;  
2.Точность не позволяет различных толкований.  
Задание 8. Перечислите значения спецификаций.  
1.Спецификации являются заданием на разработку ПО и их выполнение - закон для разработчика.  
2.Спецификации используются для проверки готовности ПО.  
3.Спецификации являются неотъемлемой частью программной документации, облегчают сопровождение и модификацию ПО.  
4.все верно.  
Задание 9. Что происходит на второй стадии?  
1.Формируется структура ПО и разрабатываются алгоритмы, задаваемые спецификациями;  
2.Устанавливается состав модулей с разделением их на иерархические уровни на основе изучения схем алгоритмов;  
3.Выбирается структура информационных массивов;  
4.Фиксируются межмодульные интерфейсы;  
5.все верно;  
6.нет правильных ответов.  
Задание 10. Что такое цель этапа?  
1. иерархическое разбиение сложных задач создания ПО на подзадачи меньшей сложности;  
2.Результатом работы на этом этапе являются спецификации на отдельные модули, дальнейшая декомпозиция которых нецелесообразна.

Код и наименование индикатора достижения компетенции	Образовательный результат
<b>ПК-1.5.1</b>	Обучающийся знает:Технические требования к пользовательскому

	интерфейсу. Стандарты регламентирующие требования к эргономике разработки взаимодействия человек-систем. Архитектуру целевой аппаратной платформы, для которой разрабатывается программное обеспечение.
--	---

<p>Задание 11. Для чего предназначен маркетинг?</p> <p>1.Результатом работы на этом этапе являются спецификации на отдельные модули, дальнейшая декомпозиция которых нецелесообразна;</p> <p>2.Изучение также существующие аналоги и продукты-конкуренты.</p> <p>Задание 12. Чем характеризуется каскадная модель?</p> <p>1. характеризуется последовательным выполнением входящих в ее состав этапов, окончанием каждого предыдущего этапа до начала последующего, отсутствием временного перекрытия этапов (последующий этап не начнется, пока не завершится предыдущий), отсутствием (или определенным ограничением) возврата к предыдущим этапам. Выявление и устранение ошибок в каскадной модели производится только на этапе тестирования;</p> <p>2.Основной ее особенностью является наличие обратных связей между этапами, проведением проверок и корректировок проектируемого программного обеспечения на каждой стадии разработки</p> <p>Задание 13. Кто такой администратор сети ?</p> <p>1.человек, который обеспечивает организационную поддержку работы локальной сети;</p> <p>2.Основной потребитель программ.</p> <p>Задание 14. Основным потребителем программ является?</p> <p>1.конечный пользователь, который, как правило, не является специалистом в области программирования;</p> <p>2.Он не программируют, а используют готовые программы для обеспечения работы на ЭВМ.</p>	
---	--

## 2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат

Код и наименование индикатора достижения компетенции	Образовательный результат
<b>УК-2.1.2</b>	Обучающийся умеет: Определять круг задач в рамках разработки программного обеспечения, планировать собственную деятельность исходя из имеющихся ресурсов.Соотносить главное и второстепенное, решать поставленные задачи в рамках разработки программного обеспечения.

<p>Задание 1. Задания выполняются в информационно-поисковой системе нормативов NormaCS. При описании хода выполнения работы указывайте, с помощью каких инструментов NormaCS выполнялось соответствующее задание. В процессе выполнения работы используйте руководство пользователя NormaCS и презентации по этой системе (входят в комплект лабораторных работ). Для работы дома подойдет сетевая lite-версия, которую можно скачать с официального сайта по адресу <a href="http://www.normacs.ru/lite.jsp">http://www.normacs.ru/lite.jsp</a>.</p> <p>1. Приведите сведения об основных элементах интерфейса системы NormaCS (в описание можно включить скриншоты).</p> <p>2. Занесите в отчет полное наименование стандартов в сфере информационных технологий следующих категорий и ссылки на них: ГОСТ, ГОСТ Р, ОСТ, ГОСТ Р ИСО/МЭК. По каждой категории достаточно привести</p>	
--	--

2-3 документа.

Пример описания хода работы: Для поиска стандартов категории ГОСТ в сфере ИТ на вкладке «Поиск» были указаны следующие параметры: «Индекс» - ГОСТ, «Область поиска» - Классификатор ISO → Информационные технологии.

Машины конторские. Результаты поиска по сформированному запросу: 2.1. ГОСТ 28470-90 Система технического обслуживания и ремонта технических средств вычислительной техники и информатики. Виды и методы технического обслуживания и ремонта ([normacs://normacs.ru/SIU?dob=42736.000347&dol=42765.596030](http://normacs.ru/SIU?dob=42736.000347&dol=42765.596030)). 2.2. ...

3. Стандарты, найденные при выполнении п.2, добавьте в панель избранных документов в NormaCS. 6

4. Найдите формулировки терминов согласно варианту (см. табл.1). Результаты оформите в виде таблицы со столбцами «Термин», «Определение», «Источник»\* , «Предметная область»\*\* .

Задание 2. Опишите жизненный цикл программного средства, разработанного в рамках выполнения курсовой работы (например, по «Технологии программирования», или «Информатике», или «Языкам программирования»), в виде списка процессов (см. рис. 1, 2) на основе стандарта ГОСТ Р ИСО/МЭК 12207-2010. Для всех выбранных процессов жизненного цикла укажите атрибуты (цель и выходы). При этом рекомендуется произвести адаптацию стандарта. Адаптация может быть выполнена по нескольким направлениям. Во-первых, можно исключить процессы, не выполнявшиеся в рамках подготовки курсовой работы. Во-вторых, можно отобрать процессы жизненного цикла, требующие адаптации, и исключить отдельные выходы.

**УК-2.1.3**

Обучающийся владеет: Выявления требований потребителей к системе и их интересов. Определения значимых показателей деятельности объекта автоматизации, на изменение которых направлен проект.

Задание 3. Оценка качества программного средства 1.1. Выберите программную продукцию. Приведите краткую информацию об оцениваемом программном средстве (название, разработчик, год создания, язык программирования).

1.2. Выберите 2-3 характеристики/подхарактеристики качества из стандарта ГОСТ Р ИСО/МЭК 25010–2015 «Модели качества систем и 19 программных продуктов». Термины и определения характеристик и подхарактеристик качества приведены в п.4 указанного стандарта. 1.3. Для выбранных характеристик/подхарактеристик по Приложению А стандарта ГОСТ Р ИСО/МЭК 25021–2014 «Элементы показателя качества» подберите 2-3 ЭПК. На примере оцениваемой программной продукции проведите измерение этих ЭПК. В отчет занесите описание метода измерения и полученные количественные значения.

Задание 4. Спроектировать жизненный цикл информационной системы предприятия с использованием спиральной модели.

**УК-2.2.2.**

Обучающийся умеет: Планировать проектные работы; выбирать методики разработки требований к системе и шаблоны документов требований к системе; формулировать цели, исходя из анализа проблем, потребностей и возможностей.

Задание 5: разработать ПС, в котором присутствовали бы некоторые критерии и примитивы качественного программного обеспечения. Сделать вывод о проделанной работе.

Вариант №1

Найдите Пифагоровы числа на заданном отрезке  $[a, b]$ .

Пример:  $3^2 + 4^2 = 5^2$

Вариант №2

Дан прямоугольник со сторонами «*a*»и «*b*», разрезать его на квадраты и посчитать их количество.

Вариант №3

Последовательность Фибоначчи 1,1,2,3,5,8,13...

Найдите *n*-ный элемент последовательности.

Вариант №4

Имеется арифметический квадрат

1 1 1 1 1 1 ...

1 2 3 4 5 6 ...

1 3 6 10 15 21...

1 4 10 20 35 56...

Найдите  $n(i,j)$  член квадрата

Вариант №5

Квадрат любого натурального числа *n* равен сумме *n* первых нечетных чисел

$$1^2=1$$

$$2^2=1+3$$

$$3^2=1+3+5$$

$$4^2=1+3+5+7$$

$$5^2=1+3+5+7+9$$

.....

Составьте ПС, позволяющую напечатать квадраты натуральных чисел от «*a*» до «*b*».

**УК-2.2.3**

Обучающийся владеет: Создавать графические документы в программах растровых и векторных изображений. Разработки блок-схемы разрабатываемых систем. Реинжиниринга разработанного программного обеспечения для решения технических задач.

Задание 6. Спроектировать жизненный цикл информационной системы предприятия с использованием каскадной модели

Задание 7. Спроектировать жизненный цикл информационной системы предприятия с использованием поэтапной модели

**ПК-1.5.2.**

Обучающийся умеет: Разрабатывать графический дизайн интерфейсов. Создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов. Осуществлять отладку программных продуктов для целевой операционной системы.

Задание 8. Спроектировать жизненный цикл информационной системы предприятия с использованием логической модели

Задание 9. Спроектировать жизненный цикл информационной системы предприятия с использованием интеллектуальной модели



ПК-1.5.3	Обучающийся владеет: Опыт применения нормативной базы и решения задач в области разработки программного обеспечения. Составления и согласования перечня требований к системе.
<p>Задание 10. Разработать ПС по автоматизации работ фирмы «Чернокнижник», занимающейся продажей книжной продукции.</p> <p>ПС должно обрабатывать информацию о книгах: форма продажи (почта, магазин, подписка и т.д.) список имеющихся в наличии книг, сгруппированных по тематике( техническая, специальная, художественная и т. д.). При этом для каждой книги указывается: серия, код, название, автор, количество экземпляров в наличии, количество проданных экземпляров, цена за единицу.</p> <p>Также ПС должно владеть информацией о заказчиках на книжную продукцию: фамилия, имя, отчество, возраст, адрес, образование, доход, сведения об оплате.</p> <p>Осуществлять подбор для каждого заказчика литературы по заданной тематике.</p> <p>Проводить анализ рынка торговли книгами:</p> <ul style="list-style-type: none"> <li>– выяснить предпочитаемые формы продажи для книг различной тематики. Общий доход от продажи книг. Книги, какой тематики пользуются наибольшей популярностью и приносят максимальный доход;</li> <li>– определить корреляционную зависимость между доходом населения, образованностью и суммой, затраченной на покупку книг;</li> </ul> <p>выяснить самый читающий слой населения. Выполнить графическую интерпретацию полученных результатов.</p>	

### 2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

#### Вопросы к экзамену

- 1 Сущность предмета ТП, его задачи. Актуальность проблемы технологии программирования. История развития ТП.
- 2 Типы ПО.
- 3 Уникальное ПО и ПО, как продукция. Требования к ПО как к продукции. Доведение ПО до товарного уровня.
- 4 Жизненный цикл ПС. Модели жизненного цикла ПС.
- 5 Водопадная модель ЖЦ ПС.
- 6 Каскадная модель ЖЦ ПС. Усовершенствование каскадной модели ЖЦ ПС.
- 7 Спиральная модель ЖЦ.
- 8 Понятие качества ПО. Критерии качества ПО: функциональность, надежность, их примитивы.
- 9 Критерии качества: легкость применения, эффективность, их примитивы.
- 10 Критерии качества: сопровождаемость, мобильность, их примитивы.
- 11 Функциональные и конструктивные критерии качества. Факторы, определяющие качество ПО.
- 12 Оценка качества ПО (показатель качества, единичный, комплексный, групповой). Методы определения числовых показателей качества.
- 13 Стиль программирования. Типы комментариев, их расположение. Выбор имен переменных. Размещение операторов. Пользовательский интерфейс (командный, графический).
- 14 Цель модульного программирования. Основные характеристики программного модуля. Размер модуля. Рутинность модуля.
- 15 Связность модуля.
- 16 Сцепление модулей.
- 17 Методы разработки структуры ПС. Восходящая разработка ПС. Архитектурный подход разработки ПС.
- 18 Нисходящая разработка ПС. Конструктивный подход разработки ПС. Метод целенаправленной конструктивной реализации.
- 19 Вспомогательные средства проектирования ПС (схемы Варнье-Орра, СИС, схемы НИРО, привести примеры).

- 20 Порядок разработки программного модуля.
- 21 Структурное программирование. Схемы передач управления.
- 22 Методы проектирования модуля: пошаговая детализация; анализ сообщений.
- 23 Методы проектирования модуля: метод расширения ядра, спецификация модуля, иерархическое проектирование модулей.
- 24 Вспомогательные средства проектирования модулей: таблицы данных, таблицы решений. Документация.
- 25 Источники ошибок в ПС: интеллектуальные возможности человека, модель перевода информации. Причины появления ошибок.
- 26 Методы обнаружения ошибок. Логические ошибки. Ошибки в числовых расчетах.
- 27 Основные понятия отладки и тестирования. Различие между отладкой и тестированием. Преимущество тестирования сверху вниз. Проверка программ в нормальных, экстремальных и исключительных ситуациях.
- 28 Основные принципы тестирования программ. Заповеди по тестированию, предложенные Г. Майерсом. Методы тестирования, два подхода к тестированию.
- 29 Тестирование модулей: тестирование путей, структур управления, ветвлений, специальных значений.
- 30 Логическая организация данных. Представление данных (внешнее, внутреннее). Физическая организация данных. Эргономические факторы при проектировании данных.
- 31 Выбор и обоснование языка программирования. Критерии выбора языка программирования.
- 32 Сравнение языков программирования (типы данных, подпрограммы, работа с динамической памятью, обработка исключительных ситуаций, параллельная обработка программных модулей и др.)
- 33 Внешнее описание ПС.
- 34 Определение требований к ПС.
- 35 Функциональная спецификация ПС. Методы контроля внешнего описания ПС.
- 36 Техническое задание на разработку ПС.
- 37 Понятие архитектуры ПС. Основные классы архитектур ПС. Контроль архитектуры ПС.
- 38 Определение основных компонентов системы: потоков данных и процессов.
- 39 Вспомогательные средства проектирования ПС (функциональные схемы, ПЕРТ-диаграмма, сети Петри). Проектная документация.
- 40 Необходимость коллективной разработки ПО.
- 41 Метод бригады главного программиста. Состав бригады.
- 42 Обязанности главного программиста.
- 43 Функции заместителя главного программиста.
- 44 Работа членов бригады. Работа секретаря (библиотекаря).
- 45 Преимущества и трудности бригадного подхода.
- 46 Проблемы оценки квалификации отдельных специалистов в коллективе.
- 47 Организация контроля при коллективной разработке программ.
- 49 Современная организация коллектива разработчиков ПС. Организация коллективов для создания очень больших комплексов программ.
- 50 Прикладное тестирование специалистов.
- 51 Документация, создаваемая и используемая в процессе разработки программных средств.
- 52 Пользовательская документация ПС.
- 53 Документация по сопровождению ПС.
- 54 Документация ПО. Стандартизация программной документации. Единая система программной документации (ЕСПД). Классификация и обозначение стандартов ЕСПД.
- 55 Назначение ЕСПД, область распространения и состав ЕСПД. Виды программных документов. Виды эксплуатационных документов. Схемы алгоритмов.
- 56 Стадии разработки программной документации.
- 57 Общие требования к программным документам.
- 58 Техническое задание. Требования к содержанию и оформлению.
- 59 Программа и методика испытаний. Текст программы, описание программы, пояснительная записка, описание применения (документация).

60 Руководство системному программисту, руководство программисту, руководство оператору. IEEE.

61 Общая характеристика состояния применения ЕСПД. Межгосударственные стандарты.

62 Надежность ПС. Показатели надежности: качественные, порядковые, количественные.

63 Факторы, определяющие надежность ПО.

64 Применение статистики к расчету надежности ПО.

65 Модели, базирующиеся на теории надежности технических систем.

66 Модель ошибок Шумана. Модель надежности.

67 Модели, сеющие предварительные ошибки.

### **Вопросы к зачету**

1. Этапы классического жизненного цикла программных систем.
2. Формы макетирования.
3. Понятие меры и метрики.
4. Выполнение оценки программного проекта.
5. Анализ риска.
6. Трассировка и контроль.
7. Размерно-ориентированные метрики.
8. Функциональный указатель.
9. Функционально-ориентированные метрики.
10. Диаграмма потоков данных.
11. Понятие активатора процесса.
12. Понятие условия данных.
13. Понятие управляющей спецификации.
14. Таблица активации процессов.
15. Диаграмма переходов-состояний.
16. Базовые элементы диаграммы Варнье.
17. Шаги метода Джексона.
18. Структурные диаграммы Джексона.
19. Структуры объектов Джексона.
20. Диаграмма системной спецификации Джексона.
21. Этапы синтеза программных систем.
22. Разработка данных.
23. Разработка архитектуры.
24. Процедурная разработка.
25. Особенности этапов проектирования.
26. Связанность модулей.
27. Сцепление модулей.
28. Оценка сложности.
29. Структурное проектирование.

### **3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации**

#### **Критерии формирования оценок по ответам на вопросы, выполнению тестовых заданий**

- оценка **«отлично»** выставляется обучающемуся, если количество правильных ответов на вопросы составляет 100 – 90% от общего объема заданных вопросов;
- оценка **«хорошо»** выставляется обучающемуся, если количество правильных ответов на вопросы – 89 – 76% от общего объема заданных вопросов;
- оценка **«удовлетворительно»** выставляется обучающемуся, если количество правильных ответов на тестовые вопросы – 75–60 % от общего объема заданных вопросов;
- оценка **«неудовлетворительно»** выставляется обучающемуся, если количество правильных ответов – менее 60% от общего объема заданных вопросов.

## Критерии формирования оценок по результатам выполнения заданий

**«Отлично/зачтено»** – ставится за работу, выполненную полностью без ошибок и недочетов.

**«Хорошо/зачтено»** – ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов.

**«Удовлетворительно/зачтено»** – ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и двух недочетов.

**«Неудовлетворительно/не зачтено»** – ставится за работу, если число ошибок и недочетов превысило норму для оценки «удовлетворительно» или правильно выполнено менее 2/3 всей работы.

*Виды ошибок:*

- *грубые ошибки: незнание основных понятий, правил, норм; незнание приемов решения задач; ошибки, показывающие неправильное понимание условия предложенного задания.*

- *негрубые ошибки: неточности формулировок, определений; нерациональный выбор хода решения.*

- *недочеты: нерациональные приемы выполнения задания; отдельные погрешности в формулировке выводов; небрежное выполнение задания.*

## Критерии формирования оценок по экзамену/зачету с оценкой

**«Отлично/зачтено»** – студент приобрел необходимые умения и навыки, продемонстрировал навык практического применения полученных знаний, не допустил логических и фактических ошибок

**«Хорошо/зачтено»** – студент приобрел необходимые умения и навыки, продемонстрировал навык практического применения полученных знаний; допустил незначительные ошибки и неточности.

**«Удовлетворительно/зачтено»** – студент допустил существенные ошибки.

**«Неудовлетворительно/не зачтено»** – студент демонстрирует фрагментарные знания изучаемого курса; отсутствуют необходимые умения и навыки, допущены грубые ошибки.

Экспертный лист  
оценочных материалов для проведения промежуточной аттестации по  
дисциплине «Технология разработки программного обеспечения»

по направлению подготовки/специальности

**09.03.01 Информатика и вычислительная техника**  
(код и наименование)

Направленность (профиль)/специализация

**Проектирование АСОИУ на транспорте**  
(наименование)

1. Формальное оценивание			
Показатели	Присутствуют	Отсутствуют	
Наличие обязательных структурных элементов:	+		
– титульный лист	+		
– пояснительная записка	+		
– типовые оценочные материалы	+		
– методические материалы, определяющие процедуру и критерии оценивания	+		
Содержательное оценивание			
Показатели	Соответствует	Соответствует частично	Не соответствует
Соответствие требованиям ФГОС ВО к результатам освоения программы	+		
Соответствие требованиям ОПОП ВО к результатам освоения программы	+		
Ориентация на требования к трудовым функциям ПС (при наличии утвержденного ПС)	+		
Соответствует формируемым компетенциям, индикаторам достижения компетенций	+		

Заключение: ФОС рекомендуется/ не рекомендуется к внедрению; обеспечивает/ не обеспечивает объективность и достоверность результатов при проведении оценивания результатов обучения; критерии и показатели оценивания компетенций, шкалы оценивания обеспечивают/ не обеспечивают проведение всесторонней оценки результатов обучения.

Эксперт, должность, ученая степень, ученое звание \_\_\_\_\_ /

(подпись)