

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таскаев Сергей Валерьевич
Должность: Ректор
Дата подписания: 2025.02.16 09:00:59
Уникальный программный ключ:
054c0182970293149c2169910009940292896864

МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю) «Программная инженерия» по направлению подготовки (специальности) 02.03.02 «Фундаментальная информатика и информационные технологии» направленности (профилю) «Математические и алгоритмические основы интеллектуальных систем» ФГБОУ ВО «ЧелГУ»

стр. 1

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю)

Программная инженерия

Направление подготовки (специальность)

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль)

Математические и алгоритмические основы интеллектуальных систем

Присваиваемая квалификация (степень)

бакалавр

Форма обучения

очная

Троицк, 2025 г.



Содержание

1. Паспорт фонда оценочных средств
2. Перечень формируемых компетенций
 - 2.1. Компетенции, закреплённые за дисциплиной
3. Содержание оценочных средств по дисциплине
 - 3.1. Виды оценочных средств
 - 3.2. Содержание оценочных средств
4. Порядок проведения и критерии оценивания промежуточной аттестации
 - 4.1. Порядок проведения промежуточной аттестации
 - 4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств
 - 4.3. Результаты промежуточной аттестации и уровни сформированности компетенций



1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Направление подготовки: *02.03.02 «Фундаментальная информатика и информационные технологии»*

Направленность (профиль): *Математические и алгоритмические основы интеллектуальных систем*

Дисциплина: *Программная инженерия.*

Семестры изучения: *7*

Формы промежуточной аттестации: *зачет с оценкой – 7 с.*

Для оценивания результатов используется балльно-рейтинговая система

2. ПЕРЕЧЕНЬ ФОРМИРУЕМЫХ КОМПЕТЕНЦИЙ

2.1. Компетенции, закреплённые за дисциплиной

Изучение дисциплины «Программная инженерия» направлено на формирование следующих компетенций:

Коды компетенции согласно ФГОС (ОПОП ВО)	Содержание компетенций согласно ФГОС (ОПОП ВО)	Индикаторы достижения компетенции согласно ОПОП	Перечень планируемых результатов обучения по дисциплине
1	2	3	4
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.	УК-1.1. Выполняет поиск информации, определяет критерии системного анализа поставленных задач	Знать. Для достижения индикатора УК-1.1: методы и средства поиска информации, определяет критерии системного анализа поставленных задач.
		УК-1.2. Использует критический анализ, систематизацию и обобщение информации для решения поставленных задач	Уметь: Для достижения индикатора УК-1.2: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач. Владеть: навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий.
ОПК-2	Способен применять компьютерные/суперкомпьютерные методы, современное программное	ОПК-2.1. Демонстрирует знание методов использования инструментальных средств, готового программного	Знать. Для достижения индикатора ОПК-2.1: методы экономического анализа производственно-хозяйственной деятельности предприятия; технологии программной инженерии; основные и вспомогательные процессы



	обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности	обеспечения и библиотек; знаком с содержанием Единого реестра российских программ	программной инженерии; преимущества инженерного подхода к созданию программного обеспечения; основные источники текущей информации по управлению ИТ-сервисами;
		ОПК-2.2. Демонстрирует умения выбирать и использовать инструментальные средства, готовое программное обеспечение и библиотеки	Уметь. Для достижения индикатора ОПК-2.2: реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; выполнять сопровождение программного обеспечения; анализировать предметную область и выделять требования к разрабатываемой программной системе; применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных задач; реализовывать проектную деятельность инструментами и методами программной инженерии;
		ОПК-2.3. Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения и сетевых коммуникаций	Владеть: Для достижения индикатора ОПК-2.3: навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий.
ПК-2	Способность к осуществлению интеграции программных модулей и компонент и проверки работоспособности программного продукта на основе международных и профессиональных стандартов информационных технологий, современных парадигм и методологий, инструментальных и вычислительных средств, методов и механизмов оценки и анализа	ПК-2.1. Обладает знаниями о методах и средствах сборки модулей и компонент программного обеспечения, о разработке процедур для развертывания программного обеспечения, миграции и преобразования данных, о создании программных интерфейсов; о методах и механизмах оценки и анализа функционирования средств и систем информационных технологий; о международных и профессиональных	Знать: Для достижения индикатора ПК-2.1: процессы жизненного цикла программного обеспечения.



	функционирования средств и систем информационных технологий.	стандартах информационных технологий, современных парадигмах методологиях, инструментальных и вычислительных средствах.	о и и
		ПК-2.2. Демонстрирует умения: применять методы и средства сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; проводить проверку и оценку работоспособности программного продукта.	Уметь. Для достижения индикатора ПК-2.2: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения.
		ПК-2.3. Имеет практический опыт (навыки): сборки модулей и компонент программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов; оценки работоспособности программного продукта.	Владеть: Для достижения индикатора ПК-2.3: навыками управления IT-проектами.
ПК-3	Способность к разработке требований и проектированию программного обеспечения на	ПК-3.1. Обладает знаниями о методах и средствах проектирования программного обеспечения, структур	Знать. Для достижения индикатора ПК-3.1: методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий; методы и средства



	основе применения базовых математических знаний и информационных технологий при решении проектно-технических и прикладных задач	данных, баз данных, программных интерфейсов.	проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.
		ПК-3.2. Демонстрирует умения: разрабатывать требования к программному продукту, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.	Уметь. Для достижения индикатора ПК-3.2: теории, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей; методы построения математических моделей, используемые при анализе предметной области; осуществлять мониторинг и оценку качества процессов производственной деятельности; понятие качества ПО, характеристики и атрибуты качества, связь атрибутов качества с требованиями;
		ПК-3.3. Имеет практический опыт (навыки): проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов.	Владеть: Для достижения индикатора ПК-3.3: навыками использования методов и механизмов оценки и анализа функционирования средств ИТ; навыками реализации проектную деятельность инструментами и методами программной инженерии.

3. СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

3.1 Виды оценочных средств

№ п/п	Контролируемые темы/ разделы	Код компетенции/ планируемые результаты обучения	Наименование оценочного средства для текущего контроля	Наименование оценочного средства на промежуточной аттестации
1	Стандарты разработки ПО	Для достижения УК-1.1. Знать: модели процесса разработки; методы и средства поиска информации, определяет критерии системного анализа поставленных задач; Для достижения УК-1.2. Уметь: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач; применять на практике критерии успешности проекта; Для достижения УК-1.2. Владеть: CASE-средствами; Для достижения ОПК-2.1. Знать:	интеллект-карта, вопросы и задания для самостоятельной работы, отчет по лабораторной работе № 1.	Интеллект-карта, кейс вопросов и заданий №1 зачетного занятия.



		<p>технологии программной инженерии; основные и вспомогательные процессы программной инженерии</p> <p>Для достижения ОПК-2.2. Уметь: давать характеристики технологиям;</p> <p>Для достижения ОПК-2.3. Владеть: StarUML;</p> <p>Для достижения ПК-2.1. Знать: процессы жизненного цикла программного обеспечения; методы управления проектами; методы работы с StarUML;</p> <p>Для достижения ПК-2.2. Уметь: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения; работать с программными средствами поддержки жизненного цикла ПО;</p> <p>Для достижения ПК-2.3. Владеть: навыками управления IT-проектами; методами работы с Designer 2000;</p> <p>Для достижения ПК-3.1. Знать: методы мониторинга и оценки процессов; методы управления проектами;</p> <p>Для достижения ПК-3.2. Уметь: работать с CASE-средствами;</p> <p>Для достижения ПК-3.3. Владеть: методами создания моделей процессов.</p>		
2	Критерии успешности проекта. Организация проектной команды	<p>Для достижения УК-1.1. Знать: модели процесса разработки; методы и средства поиска информации, определяет критерии системного анализа поставленных задач;</p> <p>Для достижения УК-1.2. Уметь: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач; применять на практике критерии успешности проекта;</p> <p>Для достижения УК-1.2. Владеть: CASE-средствами;</p> <p>Для достижения ОПК-2.1. Знать: технологии программной инженерии; основные и вспомогательные процессы</p>	интеллект-карта, вопросы и задания для самостоятельной работы, отчет по лабораторной работе №2	Интеллект-карта, кейс вопросов и заданий № 2 зачетного занятия.



		<p>программной инженерии Для достижения ОПК-2.2. Уметь: давать характеристики технологиям; Для достижения ОПК-2.3. Владеть: StarUML; Для достижения ПК-2.1. Знать: процессы жизненного цикла программного обеспечения; методы управления проектами; методы работы с StarUML; Для достижения ПК-2.2. Уметь: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения; работать с программными средствами поддержки жизненного цикла ПО; Для достижения ПК-2.3. Владеть: навыками управления IT-проектами; методами работы с Designer 2000; Для достижения ПК-3.1. Знать: методы мониторинга и оценки процессов; методы управления проектами; Для достижения ПК-3.2. Уметь: работать с CASE-средствами; Для достижения ПК-3.3. Владеть: методами создания моделей процессов.</p>		
3	Базовое расписание проекта. Управление рисками проекта	<p>Для достижения УК-1.1. Знать: методы организации проектной команды; методы и средства поиска информации, определяет критерии системного анализа поставленных задач; Для достижения УК-1.2. Уметь: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач; характеризовать жизненный цикл проекта; Для достижения УК-1.2. Владеть: CASE-средствами; Для достижения ОПК-2.1. Знать: технологии программной инженерии; основные и вспомогательные процессы программной инженерии Для достижения ОПК-2.2. Уметь: давать характеристики</p>	интеллект-карта, вопросы и задания для самостоятельной работы, отчет по лабораторной работе №3	Кейс вопросов и заданий №3 зачетного занятия



		<p>технологиям; применять методы CASE-аналитик</p> <p>Для достижения ОПК-2.3. Владеть: BPWin;</p> <p>Для достижения ПК-2.1. Знать: процессы жизненного цикла программного обеспечения; методы управления проектами; методы работы с BPWin;</p> <p>Для достижения ПК-2.2. Уметь: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения; работать с программными средствами поддержки жизненного цикла ПО; создавать диаграммы в среде StarUML;</p> <p>Для достижения ПК-2.3. Владеть: навыками управления IT-проектами; методами работы с StarUML;</p> <p>Для достижения ПК-3.1. Знать: методы мониторинга и оценки процессов; методы управления проектами;</p> <p>Для достижения ПК-3.2. Уметь: работать с CASE-средствами;</p> <p>Для достижения ПК-3.3. Владеть: методами создания моделей процессов.</p>		
4	Рабочее планирование. Принципы качественного управления	<p>Для достижения УК-1.1. Знать: модели процесса разработки; методы и средства поиска информации, определяет критерии системного анализа поставленных задач; методы инициации проекта; выполнять оценку трудоемкости и средств разработки ПО;</p> <p>Для достижения УК-1.2. Уметь: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач; применять на практике критерии успешности проекта; проектировать структуру программы;</p> <p>Для достижения УК-1.2. Владеть: CASE-средствами;</p> <p>Для достижения ОПК-2.1. Знать: технологии программной инженерии; основные и вспомогательные процессы</p>	вопросы и задания для самостоятельной работы, отчет по лабораторной работе	Кейс вопросов и заданий №4 для зачетного занятия



		<p>программной инженерии Для достижения ОПК-2.2. Уметь: давать характеристики технологиям; создавать отчеты; Для достижения ОПК-2.3. Владеть: StarUML; Для достижения ПК-2.1. Знать: процессы жизненного цикла программного обеспечения; методы управления проектами; методы работы с StarUML; Для достижения ПК-2.2. Уметь: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения; работать с программными средствами поддержки жизненного цикла ПО; кодировать и отлаживать программу; Для достижения ПК-2.3. Владеть: навыками управления IT-проектами; методами работы с Designer 2000; Для достижения ПК-3.1. Знать: методы мониторинга и оценки процессов; методы управления проектами; Для достижения ПК-3.2. Уметь: работать с CASE-средствами; выполнять реализацию и сопровождение проекта; Для достижения ПК-3.3. Владеть: методами создания моделей процессов.</p>		
--	--	---	--	--

Типовые задания, критерии и показатели оценивания в рамках текущего контроля представлены в рабочей программе по дисциплине. Полные комплекты оценочных средств и контрольно-измерительных материалов хранятся на кафедре и являются учебно-методическими материалами ограниченного (конфиденциального) пользования.

3.2 Содержание оценочных средств

Оценочные средства для промежуточной аттестации представлены базой контрольных вопросов, тестовых заданий к зачету с оценкой и практическими заданиями для зачета.

3.2.1. База тестовых вопросов



Выберите номер правильного варианта ответа

1. Жизненный цикл информационной системы– это

1. Модель создания информационной системы.
2. Модель эксплуатации информационной системы.
3. Модель проектирования информационной системы.
4. Модель создания и использования информационной системы.

2. Установите соответствие: Модель жизненного цикла

характеристика

1. Каскадная А. Делается упор на начальные этапы жизненного цикла, реализуемость технических решений проверяется путем создания прототипов.
2. Спиральная Б. Предполагает наличие циклов обратной связи между этапами, наличие межэтапных корректировок.
3. Поэтапная В. Переход на следующий этап осуществляется после полного окончания работ по предыдущему этапу.
- Г. Делается упор на последние этапы жизненного цикла, предполагается жесткая детерминация времени исполнения каждого этапа.

3. Case-средства обеспечивают...

1. Использование специальным образом организованного хранилища проектных метаданных (репозитория).
2. Сокращение персонала, связанного с информационной технологией.
3. Уменьшение степени участия в проектах высшего руководства и менеджеров, а также экспертов предметной области, уменьшение степени участия пользователей в процессе разработки приложений.
4. Немедленное повышение продуктивности деятельности организации.



4. Установите соответствие Обозначение Определение

1. DFD А. Диаграмма ключей.
2. IDEF0 Б. Диаграммы бизнес - процессов.
- В. Диаграмма потоков данных.

5. Уровнями логической модели данных являются

1. Диаграмма сущность-связь.
2. Модель данных, основанная на ключах.
3. Полная атрибутивная модель.
4. Вышеперечисленные ответы 1, 2, 3 верны.
5. Вышеперечисленные ответы 1, 3 верны.

6. Определенное свойство объекта в er-диаграмме выражает

1. Сущность.
2. Атрибут.
3. Связь.
4. Ключ.
5. Все вышеперечисленные ответы верны.

7. При установлении неидентифицирующей связи

1. Происходит миграция атрибутов первичного ключа родительской сущности в состав первичного ключа дочерней сущности.
2. Атрибуты первичного ключа родительской сущности мигрируют в состав неключевых атрибутов дочерней сущности.
3. Не происходит миграции ключей.

8. Внешний ключ – это

1. Мигрировавший в атрибуты дочерней сущности первичный ключ из родительской сущности.



2. Первичный ключ родительской сущности.
3. Первичный ключ данной сущности.
4. Атрибут, по которому возникает необходимость сортировки данных.

9. Мощность связи обозначает

1. Число дочерних сущностей у родительской .
2. Количество экземпляров дочерней сущности, связанных с одним экземпляром родительской сущности.
3. Число мигрировавших ключей.

10. Имя роли (функциональное имя) в логической модели данных – это

1. Синоним атрибута внешнего ключа в дочерней сущности.
2. Отображаемое имя связи на диаграмме.
3. Имя внешнего ключа в дочерней сущности.

11. Миграция атрибутов в логической модели данных происходит при установлении

1. Идентифицирующей связи.
2. Неидентифицирующей связи.
3. В любом из вышеперечисленных случаев.

3.2.2. Темы самостоятельных домашних работ

1. Основные понятия программной инженерии.
2. Модели и профили жизненного цикла программных средств.
3. Модели и процессы управления проектами программных средств.
4. Управление требованиями к программному обеспечению.
5. Методы проектирования программных продуктов и признаки их классификации.
6. Языки программирования и их классификация.
7. Тестирование программного обеспечения.



3.2.3. Темы для разработки интеллект-карт:

1. Сопровождение программного обеспечения.
2. Конфигурационное управление.
3. Управление программной инженерией.
4. Процесс программной инженерии.
5. Инструменты и методы программной инженерии.
6. Качество программного обеспечения.
7. Документирование программного обеспечения.
8. Техничко-экономическое обоснование проектов программных средств.

3.2.4. База вопросов для собеседования по самостоятельным домашним работам

Кейс вопросов и заданий для собеседования № 1:

1. Что такое программная инженерия?
2. Назовите дату зарождения программной инженерии как отдельной науки.
3. В чем отличие программной инженерии от информатики?
4. В чем отличие программной инженерии от системотехники?
5. Приведите примеры дисциплин информатики и программной инженерии (дисциплины не путать с учебными предметами).
6. Что такое ПО?
7. Перечислите характеристики ПО по Бруксу и кратко характеризуйте каждую.
8. Что такое процесс создания ПО?
9. Расскажите о причинах отсутствия универсального процесса разработки ПО.
10. Почему возможно и целесообразно стандартизировать процесс на уровне компании?
11. Что такое стандартный и конкретный процессы и как они соотносятся?
12. Чем отличаются между собой текущий и конкретный процессы?

Группа вопросов для собеседования № 2:

1. Дайте определение деятельности по совершенствованию процесса.
2. В чем главная трудность совершенствования процессов в компаниях?
3. Перечислите основные направления улучшения процесса.



4. Расскажите о стратегии organization pull к внедрению инноваций. Приведите примеры.
5. Расскажите о стратегии technology push к внедрению инноваций. Приведите примеры.
6. Расскажите о достоинствах, недостатках, а также возможных рисках этих стратегий.
7. Что такое модель процесса?
8. Что такое фаза процесса?
9. Что такое вид деятельности?
10. Почему нельзя отождествлять фазы и виды деятельности? Когда и по каким причинам это все таки происходит на практике?
11. В чем достоинства водопадной модели? В чем ее историческая роль? В чем ее недостатки?
12. Как в рамках водопадной модели предполагается работать с рисками?
13. Чем виток спиральной модели отличается от фазы в водопадной модели?
14. Приведите пример последовательности витков спиральной модели.

Группа вопросов для собеседования № 3:

1. В чем достоинства и недостатки спиральной модели? Каковы ограничения этой модели?
2. Как в рамках спиральной модели предполагается работать с рисками?
3. Дайте определение рабочего продукта. Приведите примеры.
4. Чем отличается рабочий продукт от компоненты ПО?
5. Расскажите, что такое нематериальный рабочий продукт.
6. Опишите, как “работает” дисциплина обязательств.
7. Приведите примеры других видов отношений между людьми.
8. Расскажите о границах применения дисциплины обязательств.
9. Что такое проект и чем он отличается от других форм организации бизнеса и производства?
10. Дайте определение архитектуре ПО. Расскажите, какие аспекты разработки задействует это понятие.
11. Расскажите о причинах множественности точек зрения при разработке ПО.
12. Как по вашему мнению, множественность точек зрения помогает или мешает разработке?
13. Перечислите и кратко прокомментируйте разные виды диаграмм UML.



Группа вопросов для собеседования № 4:

1. В чем трудность управления требованиями? При ответе на этот вопрос имейте в виду другие инженерные области и сферы бизнеса. Старайтесь отвечать на вопрос с наружи программной инженерии, а не изнутри.
2. Перечислите способы формализации требований. Под формализацией имеется в виду способ не промежуточной, а финальной фиксации.
3. Расскажите о способах и техниках «вытягивания» требований.
4. Перечислите разные виды документов, формализующих требования.
5. Расскажите об отличии функциональных и нефункциональных требований.
6. Расскажите о типовом цикле работы с требованиями.
7. Перечислите типовые ошибки при работе с требованиями.
8. Приведите примеры проблем в проектах, где нет хорошего конфигурационного управления.
9. Неформально объясните, какие задачи выполняет конфигурационное управление в проекте.
10. Дайте формальное определение конфигурационному управлению.
11. Расскажите об известном противоречии - абсолютной сохранности и удобного доступа.
12. Приведите пример артефактов проекта, которые могут «подпадать» под конфигурационное управление.
13. Приведите пример артефактов проекта, которые могут не «подпадать» под конфигурационное управление.

3.2.5. База лабораторных работ

Варианты выполнения лабораторной работы № 1

1. Задача: инвентаризация оборудования. Процессы: учет компьютерной техники, ее составных частей (клавиатур, мониторов, комплектующих и т.д.) в пределах кафедры (месторасположение, предназначение, ответственный за технику).
2. Задача: поддержание в актуальном состоянии схемы сети. Процессы: учет компьютеров и сетевого оборудования, связей между ними, и истории изменения этих связей.



3. Задача: планирование занятости аудиторий на кафедре. Процессы: ведение реестра аудиторий, характеристики аудиторий, ведение расписания занятий с контролем пересечений.

4. Задача: ведение списка студентов и сотрудников кафедры. Процессы: прием студентов, отчисление, перевод из группы в группу, то же – для сотрудников. Учесть, что студент может быть одновременно сотрудником.

5. Задача: составление учебного плана. Процессы: ведение списка предметов с учетом продолжительности курса, балансировка нагрузки на студентов.

6. Задача: выполнение и контроль лабораторных работ. Процессы: ведение списка студентов, разбиение по подгруппам и бригадам, учет выполнения работы, составления отчета, защиты отчета.

7. Задача: дела студентов. Все документы, касающиеся студента: номер студенческого, зачетка, диплом, заявления, отчеты, курсовые и т.д. Разделение: документ, который содержит оценку и влияет на дальнейшую судьбу студента (оперативный) или просто документ (справочный).

9. Задача: мониторинга торговых точек. Контроль за остатками товара в торговых точках. Мониторинг цен на товар в торговых точках. Контроль за регионально-торговыми представителями заказчика.

10. Задача: Написать программу по моделированию оплаты коммунальных услуг плательщиком.

Информационная система должна содержать следующие сведения: фамилия и инициалы плательщика, расчетный счет плательщика, расчетные счета получателя, перечисляемая сумма.

Программа должна обеспечивать выбор с помощью меню и выполнение одной из следующих функций: ввод с клавиатуры (из файла) данных о плательщике; вывод на экран информации о сумме, снятой с расчетного счета плательщика, введенного с клавиатуры; вывод на экран информации о плательщиках; осуществление сортировки по разным полям.

11. Задача: студенческая столовая. Процессы: предложить самим.



Лабораторная работа №1. StarUML Этап НАЧАЛО (Inception).

Цель работы – сформировать навыки: работы с реальными заказчиками программных систем; идентификации заинтересованных лиц и интервью с ними; анализа полученного материала; формулирования проблемы, ее актуальности и потребностей заинтересованных лиц.

Задание. Запустить проект разработки программного обеспечения согласно выданному заданию. Сформировать команду, распределить роли. Выполнить основные действия и создать артефакты, соответствующие данному этапу.

Методические указания и теоретические сведения.

RUP основан на трех ключевых идеях.

Весь ход работ направляется итоговыми целями проекта, выраженными в виде вариантов использования (use cases) — сценариев взаимодействия результирующей программной системы с пользователями или другими системами, при выполнении которых пользователи получают значимые для них результаты и услуги. Разработка начинается с выделения вариантов использования и на каждом шаге контролируется степенью приближения к их реализации.

Основным решением, принимаемым в ходе проекта, является архитектура результирующей программной системы. Архитектура устанавливает набор компонентов, из которых будет построено ПО, ответственность каждого из компонентов (т.е. решаемые им подзадачи в рамках общих задач системы), четко определяет интерфейсы, через которые они могут взаимодействовать, а также способы взаимодействия компонентов друг с другом.

Архитектура является одновременно основой для получения качественного ПО и базой для планирования работ и оценок проекта в терминах времени и ресурсов, необходимых для достижения определенных результатов. Она оформляется в виде набора графических моделей на языке UML.

Лабораторная работа №2. Этап РАЗВИТИЕ (Elaboration).

Задание. Выполнить основные действия и создать артефакты, соответствующие данному этапу разработки программного обеспечения.

Лабораторная работа №3. Этап КОНСТРУИРОВАНИЕ (Construction).

Задание. Выполнить основные действия и создать артефакты, соответствующие данному этапу разработки программного обеспечения.

Лабораторная работа №4. Этап ПЕРЕХОД (Transition).



Постановка задачи. Выполнить основные действия и создать артефакты, соответствующие данному этапу разработки программного обеспечения.

Лабораторная работа №5. Рефакторинг проекта с использованием шаблонов проектирования.

Постановка задачи. Выполнить улучшение разработанного в предыдущих лабораторных работах программного кода. Предусмотреть использование, где это необходимо, шаблонов проектирования.

Лабораторная работа №6. Управление требованиями в среде Star UML.

Постановка задачи. Необходимо оформить требования к программной системе в среде Star UML.

Лабораторная работа №7. Оценка качества программного продукта.

3.2.6. Список вопросов к зачету с оценкой (7 семестр)

1. Предпосылки и история дисциплины. Краткий обзор причин, истории возникновения и становления программной инженерии.

2. Краткая характеристика основных методов проектирования программ, сложившихся в процессе поиска решения основной проблемы программной инженерии - сокращения стоимости программного обеспечения.

3. Процесс создания программ.

4. Показатели качества программного продукта.

5. Стандартизация и стандарты Программной инженерии (SWEBOOK).

6. Понятие жизненного цикла ПО и технологических процессов его разработки.

7. Способы организации жизненного цикла ПО, каскадные и итеративные модели жизненного цикла.

8. Набор стандартов, регулирующих процессы разработки ПО в целом.

9. Сравнительные детали различных методологий разработки ПО, предлагаемые в рамках унифицированного процесса разработки Rational (RUP).

10. Экстремальное программирование (XP).

11. Методология MSF.



12. Вопросы, связанные с анализом предметной области и выделением требований к разрабатываемой программной системе.

13. Основные графические модели, используемые при анализе предметной области (диаграммы потоков данных и вариантов использования).

14. Понятие образца проектирования. Классификация образцов проектирования. Примеры образцов анализа и архитектурных стилей.

15. Понятие архитектуры ПО, влияние архитектуры на свойства ПО, а также методы оценки архитектуры.

16. Основные элементы унифицированного языка моделирования UML.

17. Основные факторы удобства использования ПО.

18. Психофизиологические особенности человека, делающие предметы удобными и неудобными для него.

19. Методика проектирования, ориентированная на удобство использования.

20. Тестирование программ. Виды и методы тестирования. Стратегии черного и белого ящика.

21. Статическое и динамическое тестирование. Инспекция кода.

22. Тестовая база, тест, тестовый случай, тестовый монитор, тестовое покрытие и инструменты его измерения.

23. Основные деятельности, входящие в компетенцию руководителей проектов.

24. Аспекты управления ресурсами, персоналом, рисками и коммуникациями проекта.

25. Особенности управления проектами по созданию ПО.

26. Понятие качества ПО, характеристики и атрибуты качества, связь атрибутов качества с требованиями.

27. Краткий обзор различных методов контроля качества ПО, с более детальным рассмотрением тестирования и проверки свойств на моделях.

28. ISO9000: система управления качеством.

29. ISO 12207: процессы качества ПО.

30. CMM: уровни зрелости процессов.

31. ISO 15504: аттестация, определение зрелости и усовершенствование процессов.

4. ПОРЯДОК ПРОВЕДЕНИЯ И КРИТЕРИИ ОЦЕНИВАНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ



4.1. Порядок проведения промежуточной аттестации

Промежуточная аттестация проводится в форме зачета с оценкой.

В ходе зачета студент предварительно проходит тест из 10 вопросов, с целью проверки базовых знаний, затем отвечает на вопросы зачетного занятия и решает задачу. Время выполнения – два академических часа. Во время выполнения можно использовать справочные материалы.

При дистанционном обучении устный опрос, в том числе защита курсовых работ, реализуется в Microsoft Teams, практические задания и письменные ответы размещаются в Moodle, ответы должны быть сданы в Moodle, тестирование осуществляется также в Moodle.

4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств

4.2.1. Критерии оценивания на зачете с оценкой

Критерии оценивания теоретического вопроса

Максимальный балл за ответ на теоретический вопрос — 10 баллов.

Отлично/ зачтено/ 9-10 баллов	Хорошо/ зачтено/ 7-8 баллов	Удовлетворительно/ зачтено/ 5-6 баллов	Неудовлетворительно /незачтено/ 0-4 балла
Высокий уровень освоения проверяемых компетенций	Средний уровень освоения проверяемых компетенций	Базовый уровень освоения проверяемых компетенций	Недостаточный уровень освоения проверяемых компетенций
Обучающийся отлично знает материал, умеет анализировать проблему и аргументировано изложить свою точку зрения, владеет достаточным для высказывания лексическим запасом, грамотно изъясняется на иностранном языке с использованием точных терминов и названий. Обучающийся	Обучающийся хорошо знает материал, умеет анализировать проблему и аргументировано изложить свою точку зрения, владеет достаточным для высказывания лексическим запасом, грамотно изъясняется на иностранном языке с использованием точных терминов и названий. Обучающийся допускает	Обучающийся знаком с материалом, владеет достаточным для высказывания лексическим запасом. Обучающийся допускает фактические и языковые ошибки, не оперирует лексическим запасом по теме.	Обучающийся не знает основных положений вопроса, не ориентируется в основных понятиях, излагает материал с трудом, с грубыми фактическими и языковыми ошибками, либо отказывается от ответов на вопросы.



практически не допускает ошибок.	незначительные ошибки.		
----------------------------------	------------------------	--	--

4.2.2. Критерии оценивания теста

Максимальный балл за тест — 20 баллов.

Оценка	Отлично/ зачтено	Хорошо/ зачтено	Удовлетворитель но/зачтено	Неудовлетворительно/ не зачтено
Баллы	18-20 баллов Правильный ответ на 9, 10 вопросов из 10	15-17 баллов Правильный ответ имеется для 7-8 вопросов	10-14 баллов Правильный ответ имеется на 5-6 вопросов	0-9 баллов Правильный ответ менее, чем на 5 вопросов.

4.2.3. Критерии оценивания решения задачи

Максимальный балл за тест — 20 баллов.

Оценка	Отлично/ зачтено	Хорошо/ зачтено	Удовлетворитель но/зачтено	Неудовлетворительно/ не зачтено
Баллы	18-20 баллов Решение корректное. Результат верный. Объяснения решения корректные, владение терминологией безупречное.	15-17 баллов Решение содержит не принципиальные ошибки, исправляемые при наводящих вопросах. Результат решения корректный.	10-14 баллов Решение содержит не принципиальные ошибки, частично исправляемые при наводящих вопросах. Результат решения корректный требует доработки и дополнительных пояснений.	0-9 баллов Полностью неверное решение. Наводящие вопросы не способствуют исправлению ошибок. Неверно используется математический аппарат.

«Зачтено» (45-60 баллов) – выставляется, если студент в полном объеме выполнил предложенное задание, программа работает без ошибок, корректно обрабатывает запросы пользователя, либо работает с незначительными легко устранимыми ошибками. «Зачтено» соответствует критериям «отлично», «хорошо», либо «удовлетворительно» таблицы п. 4.3.

«Не зачтено» (до 45 баллов) – выставляется, если студент не смог выполнить предложенное задание, не умеет создавать и работать с базами данных, допускает значительные ошибки в написании запросов. «Не зачтено» соответствует критерию «неудовлетворительно» таблицы п. 4.3.



«Отлично» (91-100 баллов) – выставляется студенту в том случае, если он:

- глубоко и правильно усвоил программный материал, последовательно, грамотно и логически стройно его излагает;
- владеет основными методами и алгоритмами решения задач;
- умеет увязывать теорию с практикой, показывает умение применять знания;
- выполнил экзаменационный тест не менее чем на 75%.

«Хорошо» (81-90 баллов) – выставляется студенту, если он:

- твердо знает программный материал, грамотно и по существу его излагает;
- владеет основными методами;
- не допускает существенных ошибок, но испытывает затруднения в выводах;
- умеет применять основные положения и приемы для решения задач;
- выполнил экзаменационный тест не менее чем на 50%.

«Удовлетворительно» (65-80 баллов) – выставляется студенту в том случае, если он:

- имеет знания только основного материала, но не умеет делать выводов;
- допускает ошибки, недостаточно правильные формулировки;
- с трудом увязывает основные положения с практикой;
- выполнил экзаменационный тест не менее чем на 25%.

«Неудовлетворительно» (до 65 баллов) – выставляется студенту в том случае, если он:

- не знает основополагающих вопросов изучаемого курса или значительной части программного материала;
- допускает ошибки, обнаруживает неумение их исправлять;
- не может увязать теорию с практикой;
- выполнил экзаменационный тест менее чем на 25%.

4.3. Результаты промежуточной аттестации и уровни сформированности компетенций

Код	Планируемые	Критерии оценивания
-----	-------------	---------------------



компетенци и	результаты обучения по дисциплине	Отлично Высокий уровень освоения проверяемых компетенций	Хорошо Средний уровень освоения проверяемых компетенций	Удовлетворитель но Базовый уровень уровня освоения проверяемых компетенций	Неудовлетворите льно Недостаточный уровень освоения проверяемых компетенций
УК-1	<p>Знать. Для достижения индикатора УК-1.1: методы и средства поиска информации, определяет критерии системного анализа поставленных задач.</p> <p>Уметь: Для достижения индикатора УК-1.2: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач.</p> <p>Владеть: навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий.</p>	<p>Знает: методы и средства поиска информации, определяет критерии системного анализа поставленных задач.</p> <p>Умеет: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач.</p> <p>Владеет: навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий</p>	<p>Знает: методы и средства поиска информации, определяет критерии системного анализа поставленных задач, но допускает несущественные ошибки</p> <p>Умеет: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач, но допускает несущественные ошибки</p> <p>Владеет: навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий, но допускает несущественные ошибки</p>	<p>Знает: в ограниченном объеме методы и средства поиска информации, определяет критерии системного анализа поставленных задач</p> <p>Умеет: в ограниченном объеме использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач.</p> <p>Владеет: в ограниченном объеме навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий</p>	<p>Не знает: методы и средства поиска информации, определяет критерии системного анализа поставленных задач.</p> <p>Не умеет: использовать критический анализ, систематизацию и обобщение информации для решения поставленных задач.</p> <p>Не владеет: навыком поиска и обработки информации о процессах производственной деятельности, связанной с созданием и использованием информационных технологий</p>
ОПК-2	<p>Знать. Для достижения индикатора ОПК-2.1: методы экономического</p>	<p>Знает: методы экономического анализа производственно-хозяйственной деятельности</p>	<p>Знает: методы экономического анализа производственно-хозяйственной деятельности</p>	<p>Знает: в ограниченном объеме методы экономического анализа производственно-</p>	<p>Не знает: методы экономического анализа производственно-хозяйственной деятельности</p>



	<p>анализа производственно-хозяйственной деятельности предприятия; технологии программной инженерии; основные и вспомогательные процессы программной инженерии; преимущества инженерного подхода к созданию программного обеспечения; основные источники текущей информации по управлению ИТ-сервисами;</p> <p>Уметь. Для достижения индикатора ОПК-2.2: реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; выполнять сопровождение программного обеспечения; анализировать предметную область и выделять требования к разрабатываемой программной системе; применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных</p>	<p>предприятия; технологии программной инженерии; основные и вспомогательные процессы программной инженерии; преимущества инженерного подхода к созданию программного обеспечения; основные источники текущей информации по управлению ИТ-сервисами;</p> <p>Умеет: реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; выполнять сопровождение программного обеспечения; анализировать предметную область и выделять требования к разрабатываемой программной системе; применять теории, методы, алгоритмы, системы и средства информационных</p>	<p>предприятия; технологии программной инженерии; основные и вспомогательные процессы программной инженерии; преимущества инженерного подхода к созданию программного обеспечения; основные источники текущей информации по управлению ИТ-сервисами, но допускает несущественные ошибки.</p> <p>Умеет: реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; выполнять сопровождение программного обеспечения; анализировать предметную область и выделять требования к разрабатываемой программной системе; применять теории, методы, алгоритмы, системы и средства информационных</p>	<p>хозяйственной деятельности предприятия; технологии программной инженерии; основные и вспомогательные процессы программной инженерии; преимущества инженерного подхода к созданию программного обеспечения; основные источники текущей информации по управлению ИТ-сервисами;</p> <p>Умеет: в ограниченном объеме реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; выполнять сопровождение программного обеспечения; анализировать предметную область и выделять требования к разрабатываемой программной системе; применять теории, методы, алгоритмы, системы и средства</p>	<p>предприятия; технологии программной инженерии; основные и вспомогательные процессы программной инженерии; преимущества инженерного подхода к созданию программного обеспечения; основные источники текущей информации по управлению ИТ-сервисами;</p> <p>Не умеет: реализовывать процессы управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий; выполнять сопровождение программного обеспечения; анализировать предметную область и выделять требования к разрабатываемой программной системе; применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных</p>
--	---	---	---	---	--



	<p>программной системе; применять теории, методы, алгоритмы, системы и средства информационных технологий при решении профессиональных задач; реализовывать проектную деятельность инструментами и методами программной инженерии;</p> <p>Владеть: Для достижения индикатора ОПК-2.3: навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий.</p>	<p>х задач; реализовывать проектную деятельность инструментами и методами программной инженерии;</p> <p>Владеет: навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий.</p>	<p>технологий при решении профессиональных задач; реализовывать проектную деятельность инструментами и методами программной инженерии, но допускает несущественные ошибки</p> <p>Владеет: навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий, но допускает несущественные ошибки</p>	<p>информационных технологий при решении профессиональных задач; реализовывать проектную деятельность инструментами и методами программной инженерии;</p> <p>Владеет: в ограниченном объеме навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий.</p>	<p>х задач; реализовывать проектную деятельность инструментами и методами программной инженерии;</p> <p>Не владеет: в навыком управления качеством производственной деятельности, связанной с созданием и использованием информационных технологий.</p>
ПК-2	<p>Знать: Для достижения индикатора ПК-2.1: процессы жизненного цикла программного обеспечения.</p> <p>Уметь: Для достижения индикатора ПК-2.2: разрабатывать и реализовывать процессы жизненного цикла</p>	<p>Знает: процессы жизненного цикла программного обеспечения.</p> <p>Умеет: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения.</p> <p>Владеет: навыками управления ИТ-проектами.</p>	<p>Знает: процессы жизненного цикла программного обеспечения, но допускает несущественные ошибки.</p> <p>Умеет: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения, но допускает несущественные ошибки</p>	<p>Знает: в ограниченном объеме процессы жизненного цикла программного обеспечения.</p> <p>Умеет: в ограниченном объеме разрабатывать и реализовывать процессы жизненного цикла программного обеспечения.</p> <p>Владеет: в ограниченном объеме навыками</p>	<p>Не знает: процессы жизненного цикла программного обеспечения.</p> <p>Не умеет: разрабатывать и реализовывать процессы жизненного цикла программного обеспечения.</p> <p>Не владеет: навыками управления ИТ-проектами.</p>



	программного обеспечения. Владеть: Для достижения индикатора ПК-2.3: навыками управления IT-проектами.		<i>Владеет:</i> навыками управления IT-проектами., но допускает несущественные ошибки	управления IT-проектами.	
ПК-3	Знать. Для достижения индикатора ПК-3.1: методы мониторинга и оценки качества процессов производственно й деятельности, связанной с созданием и использованием информационных технологий; методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов. Уметь. Для достижения индикатора ПК-3.2: применять теорию, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей; методы построения математических моделей, используемые при анализе предметной	<i>Знает:</i> методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий; методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов. <i>Умеет:</i> применять теорию, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей; методы построения математических моделей, используемые при анализе предметной	<i>Знает:</i> методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий; методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов, но допускает несущественные ошибки <i>Умеет:</i> применять теорию, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей, но допускает несущественные ошибки	<i>Знает:</i> в ограниченном объеме методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий; методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов. <i>Умеет:</i> в ограниченном объеме применять теорию, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей; <i>Владеет:</i> в ограниченном	<i>Не знает:</i> методы мониторинга и оценки качества процессов производственной деятельности, связанной с созданием и использованием информационных технологий; методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов. <i>Не умеет:</i> применять теорию, методы, системы и средства для решения практических задач в области информационных технологий с использованием современных языков, инструментальных средств, сервисов глобальных сетей; <i>Не владеет:</i> навыками использования методов и механизмов оценки и анализа



	сервисов глобальных сетей; методы построения математических моделей, используемые при анализе предметной области; осуществлять мониторинг и оценку качества процессов производственной деятельности; понятие качества ПО, характеристики и атрибуты качества, связь атрибутов качества с требованиями; Владеть: Для достижения индикатора ПК-3.3: навыками использования методов и механизмов оценки и анализа функционирования средств ИТ; навыками реализации проектную деятельность инструментами и методами программной инженерии.	области; осуществлять мониторинг и оценку качества процессов производственной деятельности; понятие качества ПО, характеристики и атрибуты качества, связь атрибутов качества с требованиями; Владеть: навыками использования методов и механизмов оценки и анализа функционирования средств ИТ; навыками реализации проектную деятельность инструментами и методами программной инженерии.	<i>Владеет:</i> навыками использования методов и механизмов оценки и анализа функционирования средств ИТ; навыками реализации проектную деятельность инструментами и методами программной инженерии., но допускает несущественные ошибки	объеме навыками использования методов и механизмов оценки и анализа функционирования средств ИТ; навыками реализации проектную деятельность инструментами и методами программной инженерии.	функционирования средств ИТ; навыками реализации проектную деятельность инструментами и методами программной инженерии.
--	---	--	--	---	---

Уровни сформированности компетенций определяются следующим образом:

1. Высокий уровень соответствует оценке “отлично” (“зачтено”), и предполагает:

- готовность к самостоятельной профессиональной деятельности;



- глубокое и правильное усвоение программного материала, последовательное, грамотное и логически стройное его изложение;

- владение основными методами и алгоритмами решения задач;
- умение строить математические модели, увязывать теорию с практикой, применять знания.

2. Средний уровень соответствует оценке “хорошо” (“зачтено”) и предполагает:

- твердое знание программного материала, его изложение грамотное и по существу;
- владение основными методами;
- отсутствие существенных ошибок, но затруднения в выводах и доказательствах;
- умение применять основные положения для решения задач.

3. Базовый уровень соответствует оценке “удовлетворительно” (“зачтено”), и предполагает:

- знания только основного материала, неумение делать выводы и проводить доказательства;
- ошибки, недостаточно правильные формулировки;
- трудное увязывание основных положений с практикой.

4. Низкий уровень соответствует оценке “неудовлетворительно” (“не зачтено”) и предполагает:

- незнание основополагающих вопросов изучаемого курса или значительной части программного материала;
- ошибки, неумение их исправлять;
- неумение увязать теорию с практикой.

