

МИНОБРНАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ  
Директор института прикладной  
математики и компьютерных наук  
А.В. Замятин  
« 14 » \_\_\_\_\_ 2023 г.



Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине  
(Оценочные средства по дисциплине)

**Объектно-ориентированный анализ и проектирование**

по направлению подготовки

**02.03.03 Математическое обеспечение и администрирование информационных систем**

Направленность (профиль) подготовки:

**DevOps-инженерия в администрировании инфраструктуры ИТ-разработки**

ОС составил(и):

д-р физ.-мат. наук, доцент,  
заведующий кафедрой программной инженерии

 А.Н. Моисеев

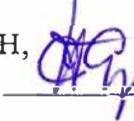
Рецензент:

д-р физ.-мат. наук, профессор,  
профессор кафедры программной инженерии

 О.А. Змеев

Оценочные средства одобрены на заседании учебно-методической комиссии  
института прикладной математики и компьютерных наук (УМК ИПМКН).

Протокол от 12.05. 2022 г. № 4

Председатель УМК ИПМКН,  С.П. Сущенко  
д-р техн. наук, профессор

**Оценочные средства (ОС)** являются элементом системы оценивания сформированности компетенций у обучающихся в целом или на определенном этапе ее формирования.

ОС разрабатываются в соответствии с рабочей программой (РП) дисциплины.

### 1. Компетенции и результаты обучения, формируемые в результате освоения дисциплины

Компетенция	Индикатор компетенции	Код и наименование результатов обучения (планируемые результаты обучения, характеризующие этапы формирования компетенций)	Критерии оценивания результатов обучения	
			Зачтено	Не зачтено
●ПК-4. Способен участвовать в разработке технической документации программных продуктов и программных комплексов	И●ПК-4.1. ●обладает необходимыми знаниями нормативной базы профессиональной деятельности	●Р-4.1.1. Знает основы унифицированного языка моделирования UML как современного профессионального стандарта информационных технологий.	●обладает необходимыми знаниями нормативной базы профессиональной деятельности Знает все аспекты унифицированного языка моделирования UML	Не обладает необходимыми знаниями нормативной базы профессиональной деятельности Не владеет основными знаниями в области языка моделирования UML, допускает грубые ошибки
	И●ПК-4.2. Применяет знания нормативной базы в профессиональной деятельности	●Р-4.2.1. Умеет применять диаграммы UML на различных этапах жизненного цикла информационных систем.	Применяет знания нормативной базы в профессиональной деятельности Умеет безошибочно применять диаграммы UML на различных этапах жизненного цикла информационных систем	Не применяет знания нормативной базы в профессиональной деятельности Не умеет применять диаграммы UML на этапах жизненного цикла информационных систем, допускает грубые ошибки

	И●ПК-4.3. Разрабатывает техническую документацию, связанную с профессиональной деятельностью	●Р-4.3.1. Владеет навыками создания диаграмм UML различных видов.	Разрабатывает техническую документацию, связанную с профессиональной деятельностью Владеет безупречными навыками создания диаграмм UML различных видов.	Не разрабатывает техническую документацию, связанную с профессиональной деятельностью Не владеет основными навыками создания диаграмм UML, допускает грубые ошибки
ПК-1. Способен осуществлять программирование, тестирование и опытную эксплуатацию ИС с использованием технологических и функциональных стандартов, современных моделей и методов оценки качества и надежности программных средств	ИПК-1.1. ●определяет, согласовывает и утверждает требования заказчика к ИС	●Р-1.1.1. Знает основы объектно-ориентированного анализа и проектирования как современной методологии разработки программного обеспечения.	●определяет, согласовывает и утверждает требования заказчика к ИС Полностью освоил теоретический материал по основам объектно-ориентированного анализа и проектирования	Не определяет, согласовывает и утверждает требования заказчика к ИС Не освоил теоретический материал по основам объектно-ориентированного анализа и проектирования и допускает грубые ошибки
	ИПК-1.2. Проектирует программное обеспечение	●Р-1.2.1. Умеет применять приемы и паттерны объектно-ориентированного анализа и проектирования.	Проектирует программное обеспечение Умеет безошибочно применять приемы и паттерны объектно-ориентированного анализа и проектирования.	Не проектирует программное обеспечение Не умеет применять основные приемы и паттерны объектно-ориентированного анализа и проектирования, допускает грубые ошибки

	<p>ИПК-1.3. Кодирует на языках программирования и проводит модульное тестирование ИС</p>	<p>●Р-1.3.1. Владеет программными средствами поддержки (автоматизации) объектно-ориентированного проектирования.</p>	<p>Кодирует на языках программирования и проводит модульное тестирование ИС Умеет с помощью программных средств поддержки (автоматизации) решать задачи объектно-ориентированного проектирования.</p>	<p>Не кодирует на языках программирования и проводит модульное тестирование ИС Не умеет с помощью программных средств поддержки (автоматизации) решать задачи объектно-ориентированного проектирования, допускает грубые ошибки.</p>
--	--	--	---	--

## 2. Этапы формирования компетенций и виды оценочных средств

№	Этапы формирования компетенций (разделы дисциплины)	Код и наименование результатов обучения	Вид оценочного средства (тесты, задания, кейсы, вопросы и др.)
1.	Язык UML	ОР-4.1.1, ОР-4.2.1, ОР-4.3.1	Контрольная работа Задания
2.	Паттерны проектирования	ОР-1.1.1, ОР-1.2.1, ОР-1.3.1	Контрольная работа Задания
3.	Архитектурные решения	ОР-1.1.1, ОР-1.2.1, ОР-1.3.1	Контрольная работа Задания

## 3. Типовые контрольные задания или иные материалы, необходимые для оценки образовательных результатов обучения

3.1. Типовые задания для проведения текущего контроля успеваемости по дисциплине. Контрольная работа № 1.

1. Пояснить смысл термина объектно-ориентированного программирования, привести примеры (по вариантам): методология объектно-ориентированного программирования, класс, экземпляр класса, инстанцирование, указатель, инкапсуляция, полиморфизм, наследование, предок / потомок, сигнатура, интерфейс, поле, метод, конструктор, переопределение метода, виртуальный метод, абстрактный класс, абстрактный метод, модификатор доступа, структура класса / объекта, состояние объекта, поведение класса / объекта, статический метод (метод уровня класса), коллекция, реализация интерфейса.
2. Задача. По заданной диаграмме классов или последовательностей выполнить прямое проектирование (записать код).
3. Задача. По заданному программному коду, выполнив обратное проектирование, создать диаграмму классов или диаграмму последовательностей.
4. Описать элементы и основы конструирования диаграмм UML (по вариантам): диаграммы классов, диаграммы последовательностей, диаграммы объектов, диаграммы коммуникаций, диаграммы пакетов, диаграммы развертывания, диаграммы вариантов использования, диаграммы состояний, диаграммы деятельности, диаграммы компонентов, диаграммы анализа.

Контрольная работа № 2.

1. Описать шаблон GRASP или паттерн (по вариантам): Слабое связывание, Сильное зацепление, Информационный эксперт, Контроллер, Создатель, Фабричный метод, Абстрактная фабрика, Строитель, Прототип, Одиночка, Адаптер, Мост, Компоновщик, Декоратор, Фасад, Заместитель, Приспособленец, Шаблонный метод, Цепочка обязанностей, Команда, Итератор, Посредник, Хранитель, Наблюдатель, Состояние, Стратегия, Посетитель.

Контрольная работа № 3.

1. Описать архитектурный паттерн общего назначения (по вариантам): Шлюз, Преобразователь, Супертип слоя, Отделенный интерфейс, Реестр, Объект-значение, Частный случай, Дополнительный модуль, Фиктивная служба, Множество записей, Сценарий транзакции, Модель предметной области, Модуль таблицы.
2. Описать архитектурное решение, связанное с базами данных (по вариантам): Поле идентификации, Отображение внешних ключей, Отображение с помощью таблицы ассоциаций, Отображение зависимых объектов, Внедренное значение, Сериализованный крупный объект, Наследование с одной таблицей, Наследование с таблицами для каждого класса, Наследование с таблицами для каждого конкретного класса, Отображение метаданных, «База данных – хранилище объектов», Шлюз таблицы данных, Шлюз записи данных, Активная запись, Преобразователь данных, Единица работы, Коллекция объектов, Загрузка по требованию, Объект запроса, Хранилище.
3. Описать архитектурное решение для построения Web-приложений или для распределенной обработки данных и задач автономного параллелизма (по вариантам): Модель–представление–контроллер, Контроллер страниц, Контроллер запросов, Представление по шаблону, Представление с преобразованием, Двухэтапное представление, Контроллер приложения, Интерфейс удаленного доступа, Объект переноса данных, Оптимистическая автономная блокировка, Пессимистическая автономная блокировка, Блокировка с низкой степенью детализации, Неявная блокировка.

### 3.2. Типовые задания для проведения промежуточной аттестации по дисциплине.

Теоретические вопросы к зачетам.

1. Описать элементы и основы конструирования диаграмм классов UML.
2. Описать элементы и основы конструирования диаграмм последовательностей UML.
3. Описать элементы и основы конструирования диаграмм объектов UML.
4. Описать элементы и основы конструирования диаграмм коммуникаций UML.
5. Описать элементы и основы конструирования диаграмм пакетов UML.
6. Описать элементы и основы конструирования диаграмм развертывания UML.
7. Описать элементы и основы конструирования диаграмм вариантов использования UML.
8. Описать элементы и основы конструирования диаграмм состояний UML.
9. Описать элементы и основы конструирования диаграмм деятельности UML.
10. Описать элементы и основы конструирования диаграмм компонентов UML.
11. Описать элементы и основы конструирования диаграмм анализа.
12. Описать паттерн проектирования Фабричный метод.
13. Описать паттерн проектирования Абстрактная фабрика.
14. Описать паттерн проектирования Строитель.
15. Описать паттерн проектирования Прототип.
16. Описать паттерн проектирования Одиночка.
17. Описать паттерн проектирования Адаптер.
18. Описать паттерн проектирования Мост.
19. Описать паттерн проектирования Компоновщик.
20. Описать паттерн проектирования Декоратор.
21. Описать паттерн проектирования Фасад.
22. Описать паттерн проектирования Заместитель.
23. Описать паттерн проектирования Приспособленец.

24. Описать паттерн проектирования Шаблонный метод.
25. Описать паттерн проектирования Цепочка обязанностей.
26. Описать паттерн проектирования Команда.
27. Описать паттерн проектирования Итератор.
28. Описать паттерн проектирования Посредник.
29. Описать паттерн проектирования Хранитель.
30. Описать паттерн проектирования Наблюдатель.
31. Описать паттерн проектирования Состояние.
32. Описать паттерн проектирования Стратегия.
33. Описать паттерн проектирования Посетитель.
34. Описать шаблон проектирования GRASP Слабое связывание.
35. Описать шаблон проектирования GRASP Сильное зацепление.
36. Описать шаблон проектирования GRASP Информационный эксперт.
37. Описать шаблон проектирования GRASP Контроллер.
38. Описать шаблон проектирования GRASP Создатель.

1. Описать архитектурный паттерн Шлюз.
2. Описать архитектурный паттерн Преобразователь.
3. Описать архитектурный паттерн Супертип слоя.
4. Описать архитектурный паттерн Отделенный интерфейс.
5. Описать архитектурный паттерн Реестр.
6. Описать архитектурный паттерн Объект-значение.
7. Описать архитектурный паттерн Частный случай.
8. Описать архитектурный паттерн Дополнительный модуль.
9. Описать архитектурный паттерн Фиктивная служба.
10. Описать архитектурный паттерн Множество записей.
11. Описать типовой прием организации бизнес-логики Сценарий транзакции.
12. Описать типовой прием организации бизнес-логики Модель предметной области.
13. Описать типовой прием организации бизнес-логики Модуль таблицы.
14. Описать архитектурное решение Поле идентификации.
15. Описать архитектурное решение Отображение внешних ключей.
16. Описать архитектурное решение Отображение с помощью таблицы ассоциаций.
17. Описать архитектурное решение Отображение зависимых объектов.
18. Описать архитектурное решение Внедренное значение.
19. Описать архитектурное решение Сериализованный крупный объект.
20. Описать архитектурное решение Наследование с одной таблицей.
21. Описать архитектурное решение Наследование с таблицами для каждого класса.
22. Описать архитектурное решение Наследование с таблицами для каждого конкретного класса.
23. Описать архитектурное решение Отображение метаданных.
24. Описать архитектурное решение «База данных – хранилище объектов».
25. Описать архитектурное решение Шлюз таблицы данных.
26. Описать архитектурное решение Шлюз записи данных.
27. Описать архитектурное решение Активная запись.
28. Описать архитектурное решение Преобразователь данных.
29. Описать архитектурное решение Единица работы.
30. Описать архитектурное решение Коллекция объектов.
31. Описать архитектурное решение Загрузка по требованию.
32. Описать архитектурное решение Объект запроса.
33. Описать архитектурное решение Хранилище.
34. Описать архитектурное решение Модель–представление–контроллер.
35. Описать архитектурное решение Контроллер страниц.

36. Описать архитектурное решение Контроллер запросов.
37. Описать архитектурное решение Представление по шаблону.
38. Описать архитектурное решение Представление с преобразованием.
39. Описать архитектурное решение Двухэтапное представление.
40. Описать архитектурное решение Контроллер приложения.
41. Описать архитектурное решение Интерфейс удаленного доступа.
42. Описать архитектурное решение Объект переноса данных.
43. Описать архитектурное решение Оптимистическая автономная блокировка.
44. Описать архитектурное решение Пессимистическая автономная блокировка.
45. Описать архитектурное решение Блокировка с низкой степенью детализации.
46. Описать архитектурное решение Неявная блокировка.

#### **4. Методические материалы, определяющие процедуры оценивания образовательных результатов обучения**

4.1. Методические материалы для оценки текущего контроля успеваемости по дисциплине.

Текущий контроль по лабораторным работам осуществляется в виде проверки выполнения заданий лабораторной работы. Текущий контроль успеваемости по теоретическому материалу осуществляется в виде контрольных работ.

Оценка текущего контроля проводится на основе оценки компетенций, соответствующих текущему разделу дисциплины, согласно таблице раздела 1.

4.2. Методические материалы для проведения промежуточной аттестации по дисциплине.

Итоговая оценка по предмету (зачет) выставляется следующим образом:

«зачтено» – студент сдал все лабораторные и контрольные работы семестра на «зачтено»;

«не зачтено» – студент не сдал хотя бы одну лабораторную или контрольную работу на «зачтено».

Во время зачета студент может улучшить свою оценку, сдав заново соответствующую контрольную работу, при условии, что он заранее сдал все лабораторные работы семестра.