Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 23.05.2024 10:24:59
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735c18b1d6

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Факультет Информационные технологии

УТВЕРЖДАЮ
Декан факультета
«Информационные технологии»
_____ / Д.Г.Демидов /
«15» февраля 2024г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Шаблоны проектирования приложений»

Направление подготовки/специальность
09.03.01 Информатика и вычислительная техника

Профиль/специализация
«Веб технологии»

Квалификация
бакалавр

Формы обучения
Очная, заочная

Москва, 2024 г.

Разработчик(и):

Доцент, к.т.н. / В.Ю. Верещагин /

Согласовано:
Заведующий кафедрой «Инфокогнитивные технологии»,
к.т.н., доцент / Е.А. Пухова /

Содержание

1 Цели, задачи и планируемые результаты обучения по дисциплине 4
2 Место дисциплины в структуре образовательной программы 6
3 Структура и содержание дисциплины 7
3.1 Виды учебной работы и трудоемкость для очной формы обучения 7
3.2 Тематический план изучения дисциплины для очной формы обучения 7
3.3 Содержание дисциплины 8
3.4 Тематика семинарских/практических и лабораторных занятий 8
4 Учебно-методическое и информационное обеспечение 9
4.1 Нормативные документы и ГОСТы 9
4.2 Основная литература 9
4.3 Дополнительная литература 10
4.4 Электронные образовательные ресурсы 10
4.5 Лицензионное и свободно распространяемое программное обеспечение 10
4.6 Современные профессиональные базы данных и информационные справочные системы 11
5 Материально-техническое обеспечение 11
6 Методические рекомендации 11
6.1 Методические рекомендации для преподавателя по организации обучения 11
6.2 Методические указания для обучающихся по освоению дисциплины 12
7 Фонд оценочных средств 12
7.1 Методы контроля и оценивания результатов обучения 12
7.2 Шкала и критерии оценивания результатов обучения 13
7.3 Оценочные средства 15

1 Цели, задачи и планируемые результаты обучения по дисциплине

К основным целям освоения дисциплины «Шаблоны проектирования приложений» относится:

получение знаний об общих принципах построения информационных систем по принципам ООП;

получение знание и умений проектирования информационных систем в соответствии с требованиями оптимизации и номенклатуре;

овладение общей методикой применения шаблонов проектирования;

закрепление получаемых в семестре знаний и навыков на практике;

формирование взаимосвязей, получаемых в семестре знаний и навыков с изученными ранее и изучаемых параллельно с данной дисциплиной;

подготовка студентов к деятельности в соответствии с квалификационной характеристикой бакалавра.

К основным задачам дисциплины «Шаблоны проектирования приложений» относятся:

получение знаний о методах применения шаблонов проектирования, об их особенностях внедрения в ИС и их функциональности, о соответствии принципов и парадигм ООП и применении шаблонов проектирования, о перспективах развития систем управления предприятиями, а также приобретение студентами практических навыков по проектированию и программированию шаблонов проектирования.

изучение и освоение теоретического материала, как в процессе контактной, так и в ходе самостоятельной работы;

выполнение предоставленных практических заданий различных форм, как в процессе контактной, так и в ходе самостоятельной работы;

самостоятельная работа над тематикой дисциплины для формирования компетенций основной профессиональной образовательной программы (далее, ОПОП).

Обучение по дисциплине «Шаблоны проектирования приложений» направлено на формирование у обучающихся следующих компетенций:

Код компетенции	**Наименование компетенции**	**Индикаторы планируемых результатов обучения по дисциплине**
ПК-1	Способность выполнять работы по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	**ИПК-1.1** Знает: методологию и технологии проектирования информационных систем; проектирование обеспечивающих подсистем; приемы программирования приложений. **ИПК-1.2.** Умеет: создавать, модифицировать и сопровождать информационные системы для решения задач бизнес-процессов и организационного управления; проводить сравнительный анализ и выбор ИКТ для решения прикладных задач и создания информационных систем; разрабатывать сайты, мобильные приложения, позволяющие автоматизировать конкретные бизнес-процессы для заданной организации.

		ИПК-1.3. Владеет: методами создания и сопровождения информационных систем, автоматизирующих задачи организационного управления и бизнес-процессы продукта; методологией и технологией проектирования информационных систем, проектирования обеспечивающих подсистем; навыками работы с web - технологиям и программировать.
ПК-3	Способность разрабатывать требования и проектировать программное обеспечение	**ИПК-3.1. Знает** возможности существующей программно-технической архитектуры; возможности современных и перспективных средств разработки программных продуктов, технических средств; методологии разработки программного обеспечения и технологии программирования; методологии и технологии проектирования и использования баз данных; языки формализации функциональных спецификаций; методы и приемы формализации задач; методы и средства проектирования программного обеспечения; методы и средства проектирования программных интерфейсов; методы и средства проектирования баз данных; принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методы и средства проектирования программного обеспечения и баз данных; методы и средства проектирования программных интерфейсов. **ИПК-3.2.** Умеет проводить анализ исполнения требований; вырабатывать варианты реализации требований; проводить оценку и обоснование рекомендуемых решений; осуществлять коммуникации с заинтересованными сторонами; выбирать средства реализации требований к программному обеспечению; вырабатывать варианты реализации программного обеспечения; проводить оценку и обоснование рекомендуемых решений; осуществлять коммуникации с заинтересованными сторонами; использовать существующие типовые решения и шаблоны проектирования программного обеспечения; применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов; осуществлять коммуникации с заинтересованными сторонами. **ИПК-3.3. Владеет** современным инструментарием и средами разработки программного кода; современным инструментарием и средами проектирования программного кода, методами тестирования ПО.

ПК-5	Способность осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности	**ИПК-5.1. Знает:** теоретические основы проектирования сайтов и Internet-приложений; стандарты оформления технических заданий; базовые технологии разработки веб-приложения на стороне клиента и стороне сервера; методы концептуального, функционального и логического проектирования систем; методы тестирования; международные стандарты на структуру документов требований; нормативные и методические материалы по созданию документов требований к системам методы оценки качества программных систем; способы масштабирования информационных систем для учета их при логическом проектировании. **ИПК5.2. Умеет:** формулировать цели, исходя из анализа проблем, потребностей и возможностей; разрабатывать технико-экономическое обоснование; декомпозировать функции на подфункции; алгоритмизировать деятельность; разрабатывать структуры типовых документов; исполнять ручные тесты, проектировать Web-сайты и разрабатывать Internet-приложения; использовать основные приемы web-дизайна. Внедрять графические. Звуковые, анимационные объекты в страницу; формировать интерактивные блоки web-ресурса; разрабатывать модели концептуальной, функциональной и логической архитектуры системы; спроектировать информационную систему для заданного предприятия по заданным характеристикам и создать ее на платформе 1С с помощью конфигурирования и программирования. **ИПК-5.3. Владеет:** навыками концептуального, функционального и логического проектирования; средствами автоматизации проектирования ПО, работы со средствами Internet и Web-технологий для решения задач профессиональной деятельности; навыками проектирования схемы последовательностей, состояний и взаимодействий компонентов системы; навыками работы в 1С Конфигуратор и программирования на платформе 1С.

2 Место дисциплины в структуре образовательной программы

Дисциплина относится к части, формируемой участниками образовательных отношений блока Б1.2.

Дисциплина взаимосвязана логически и содержательно-методически со следующими дисциплинами и практиками ОПОП:

- Проектирование бизнес-процессов и структур в веб-индустрии;

- Основы разработки мобильных приложений;
- Разработка веб-приложений;
- Разработка мобильных приложений;
- Технологии компьютерной графики.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетных единицы (108 часов).

3.1 Виды учебной работы и трудоемкость для очной формы обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			4	
1	**Аудиторные занятия**	**54**	**54**	
	В том числе:			
1.1	Лекции	18	18	
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	36	36	
2	**Самостоятельная работа**	**54**	**54**	
3	**Промежуточная аттестация**			
	Зачет/диф. Зачет/экзамен	**зачет**	**зачет**	
	Итого:	**108**	**108**	

3.2 Тематический план изучения дисциплины для очной формы обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/практические занятия	Лабораторные занятия	Практическая подготовка	
1	Паттерн Стратегия	9	1		4		4
2	Паттерн Наблюдатель	9	1		4		4
3	Паттерн Декоратор	9	1		4		4
4	Паттерн Фабрика	9	1		4		4
5	Паттерн Одиночка	9	1		4		4
6	Паттерн Команда	9	1		4		4
7	Паттерны «Адаптер» и «Фасад»	9	1		4		4
8	Паттерн «Шаблонный метод»	9	1		4		4
9	Паттерны «Итератор» и «Компоновщик»	10	3		4		6

10	Паттерн «Состояние»	10	3				6
11	Паттерн «Заместитель»	16	4				10
Итого		**108**	**18**		**36**		**54**

3.3 Виды учебной работы и трудоемкость для заочной формы обучения

№ п/п	Вид учебной работы	Количество часов	Семестры	
			6	
1	**Аудиторные занятия**	**12**	**12**	
	В том числе:			
1.1	Лекции	4	4	
1.2	Семинарские/практические занятия			
1.3	Лабораторные занятия	8	8	
2	**Самостоятельная работа**	**96**	**96**	
3	**Промежуточная аттестация**			
	Зачет/диф. Зачет/экзамен	**зачет**	**зачет**	
	Итого:	**108**	**108**	

3.4 Тематический план изучения дисциплины для заочной формы обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	Паттерн Стратегия	15	1		4		10
2	Паттерн Наблюдатель	15	1		4		10
3	Паттерн Декоратор	11	1				10
4	Паттерн Фабрика	11	1				10
5	Паттерн Одиночка	10					10
6	Паттерн Команда	10					10
7	Паттерны «Адаптер» и «Фасад»	10					10
8	Паттерн «Шаблонный метод»	10					10
9	Паттерны «Итератор» и «Компоновщик»	10					10
10	Паттерн «Состояние»	3					3
11	Паттерн «Заместитель»	3					3
Итого		**108**	**4**		**8**		**96**

3.5 Содержание дисциплины

1. Паттерн Стратегия

1.1. Основы программирования по методикам ООП, оптимизация реализации алгоритмов, инкапсуляция

2. Паттерн Наблюдатель

2.1. Абстракция, Композиция классов, Интерфейсы ООП, Методика “Low Coupling”

3. Паттерн Декоратор

3.1. Расширяемость классов, Субклассирование, Расширение функциональности, Наследование, «Прозрачность» кода

4. Паттерн Фабрика

4.1. Абстракция кода, Взаимосвязь объектов, Интерфейс создания объекта, Экземпляры объекта, Делегирование субклассам

5. Паттерн Одиночка

5.1. Одиночные экземпляры, Оптимизация производительности приложения, Многопоточность

6. Паттерн Команда

6.1. Инкапсуляция запросов, параметризация объектов, реализация очереди, Реализация «отмены»

7. Паттерны «Адаптер» и «Фасад»

7.1. Группирование интерфейсов подсистемы, преобразование интерфейсов, Изоляция клиента, Адаптеры объектов и адаптеры классов.

8. Паттерн «Шаблонный метод»

8.1. Определение основного алгоритма программы, Абстрактные методы, Инкапсуляция алгоритмов

9. Паттерны «Итератор» и «Компоновщик»

9.1. Реализация перебора элементов, Реализация древовидной структуры, Инкапсуляция коллекций, Выполнений однородных операций

10. Паттерн «Состояние»

10.1. Управление поведением объекта, Конечные автоматы, Делегирование выполняемых операций

11. Паттерн «Заместитель»

11.1. Суррогация объектов, Увеличение количества классов в системе

3.6 Тематика семинарских/практических и лабораторных занятий

1. Разработка информационной системы с применением алгоритмиов нечёткой логики.

2. Разработка информационной системы с внедрением геоинформационных данных и связью с мобильным приложением.

3. Разработка информационной системы с открытым клиенто-ориентированным дизайном.

4. Представление проекта

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный закон от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями);

2. Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденный Приказом Министерства образования и науки РФ от 19 сентября 2017 г. N 929 "Об утверждении федерального… Редакция с изменениями N 1456 от 26.11.2020

3. Приказ Министерства образования и науки РФ от 05 апреля 2017 г. № 301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры;

4. Порядок проведения государственной итоговой аттестации по образовательным программам высшего образования – программам бакалавриата, программам специалитета и программам магистратуры, утвержденный приказом Минобрнауки России от 29 июня 2015 г. № 636;

5. Положение о практической подготовке обучающихся, утвержденное приказом Министерства науки и высшего образования Российской Федерации и Министерства просвещения Российской Федерации от 5 августа 2020 г. № 885/390.

4.2 Основная литература

1. Чернышев, С. А. Принципы, паттерны и методологии разработки программного обеспечения : учебное пособие для вузов / С. А. Чернышев. — Москва : Издательство Юрайт, 2023. — 176 с. — (Высшее образование). — ISBN 978-5-534-14383-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/520097 (дата обращения: 25.09.2023).

2. Гутгарц, Р. Д. Проектирование автоматизированных систем обработки информации и управления : учебное пособие для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/509638 (дата обращения: 25.09.2023).

3. Григорьев, М. В. Проектирование информационных систем : учебное пособие для вузов / М. В. Григорьев, И. И. Григорьева. — Москва : Издательство Юрайт, 2023. — 318 с. — (Высшее образование). — ISBN 978-5-534-01305-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/512729 (дата обращения: 25.09.2023).

4. Грекул, В. И. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — 2-е изд. — Москва : Издательство Юрайт, 2023. — 423 с. — (Профессиональное образование). — ISBN 978-5-534-17836-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/533817 (дата обращения: 25.09.2023).

5. Крючкова, Е. Н. Объектно-ориентированное программирование: Архитектурное проектирование и паттерны программирования : учебно-методическое пособие / Е. Н. Крючкова, С. М. Старолетов. — Барнаул : АлтГТУ, 2020. — 180 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

https://e.lanbook.com/book/292790 (дата обращения: 25.09.2023). — Режим доступа: для авториз. пользователей.

6. Архитектурные решения информационных систем / А. И. Водяхо, Л. С. Выговский, В. А. Дубенецкий, В. В. Цехановский. — 3-е изд., стер. — Санкт-Петербург : Лань, 2023. — 356 с. — ISBN 978-5-507-46063-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/296981 (дата обращения: 25.09.2023). — Режим доступа: для авториз. пользователей.

7. Згуральская, Е. Н. Технологии программирования : учебное пособие / Е. Н. Згуральская. — Ульяновск : УлГТУ, 2020. — 71 с. — ISBN 978-5-9795-1995-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/165011 (дата обращения: 25.09.2023). — Режим доступа: для авториз. пользователей.

8. Унгер, А. Ю. Шаблоны объектно-ориентированного проектирования в языке C++ : учебное пособие / А. Ю. Унгер. — Москва : РТУ МИРЭА, 2022. — 67 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/310838 (дата обращения: 25.09.2023). — Режим доступа: для авториз. пользователей.

4.3 Дополнительная литература

1. Пикус, Ф. Идиомы и паттерны проектирования в современном C++ : руководство / Ф. Пикус ; перевод с английского А. А. Слинкина. — Москва : ДМК Пресс, 2020. — 452 с. — ISBN 978-5-97060-786-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/140598 (дата обращения: 29.09.2023). — Режим доступа: для авториз. пользователей

2. Паттерны проектирования на платформе .NET. С.Тепляков, 2015г.

3. Конкурентность и параллелизм на платформе .NET. Паттерны эффективного проектирования, Т.Рикардо, 2021г.

4. Паттерны проектирования для C# и платформы .NET Core, Ч.Джеффри, Ар.Гаурав, 2021г.

5. Паттерны объектно-ориентированного программирования. Э.Гамма, Р.Хелм, Р.Джонсон, Дж. Влиссидес, 2021г.

6. Распределённые системы. Паттерны проектирования. Б.Брендан, 2020г.

7. Паттерны проектирования API, Джей Джей Гивакс, 2023г.

4.4 Электронные образовательные ресурсы

ЭОР Паттерны проектирования приложений
https://lms.mospolytech.ru/course/view.php?id=13059

4.5 Лицензионное и свободно распространяемое программное обеспечение

Для выполнения лабораторных работ и самостоятельной работы необходимо следующее программное обеспечение:

1. IDE для разработки клиентской части приложения
2. IDE для разработки серверной части приложения
3. Веб-браузер для реализации UML-нотации.

Для проведения лекционных и практических занятий специального программного обеспечения для освоения дисциплины не требуется.

4.6 Современные профессиональные базы данных и информационные справочные системы

1. https://doka.guide/
2. https://developer.mozilla.org/ru/
3. https://roadmap.sh/frontend
4. https://learn.javascript.ru/

Информационные справочные системы

1. https://elibrary.ru - Научная электронная библиотека eLIBRARY.RU (ресурсы открытого доступа)
2. https://www.rsl.ru - Российская Государственная Библиотека (ресурсы открытого доступа)
3. https://link.springer.com - Международная реферативная база данных научных изданий Springerlink (ресурсы открытого доступа)
4. https://zbmath.org - Международная реферативная база данных научных изданий zbMATH (ресурсы открытого доступа)

Интернет-ресурсы

1. http://window.edu.ru - Информационная система "Единое окно доступа к образовательным ресурсам"
2. https://openedu.ru - «Национальная платформа открытого образования» (ресурсы открытого доступа)

5 Материально-техническое обеспечение

Для проведения лабораторных работ и самостоятельной работы студентов подходят аудитории, оснащенные компьютерами с программным обеспечением в соответствии со списком в пункте 4.5 и подключенные к интернету.

Число рабочих мест в аудитории должно быть достаточным для обеспечения индивидуальной работы студентов.

Рабочее место преподавателя должно быть оснащено компьютером с подключенным к нему проектором или иным аналогичным по функциональному назначению оборудованием.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

1. При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы – теоретические сведения, задачи и др. При проведении занятия следует контролировать подачу материала и решение заданий с учетом учебного времени, отведенного для занятия.

2. При проверке работ и отчетов следует учитывать не только правильность выполнения заданий, но и оптимальность выбранных методов решения, правильность выполнения всех его шагов.

6.2 Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи учебным планом. Основой теоретической подготовки студентов являются лекции и самостоятельная работа.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, готовятся к промежуточной аттестации, а также самостоятельно изучают отдельные темы учебной программы.

На занятиях студентов, в том числе предполагающих практическую деятельность, осуществляется закрепление полученных, в том числе и в процессе самостоятельной работы, знаний. Особое внимание обращается на развитие умений и навыков установления связи положений теории с профессиональной деятельностью будущего специалиста.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).

Текущий контроль осуществляется на аудиторных занятиях.

Критериями оценки результатов самостоятельной работы студента являются:

- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность компетенций;
- оформление материала в соответствии с требованиями.

Приветствуется обсуждение самих заданий с другими студентами: можно как давать, так и получать советы по общей стратегии выполнения и изучения материала, давать и получать помощь в отладке. Однако писать код студент должен самостоятельно. Делиться кодом или писать его совместно запрещено.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

Приведенные ниже правила выставления оценок и опозданий могут быть изменены, если преподаватель сочтет это необходимым. Важно, чтобы студенты регулярно просматривали план курса, выложенный в СДО, на предмет его обновления или изменения.

Достижение компетенций оценивается с помощью лабораторных работ и рубежных контролей. Индикаторы ИПК-1.1, ИПК-1.2, ИПК-1.3, ИПК-2.1, ИПК-2.2, ИПК-2.3, ИПК-5.1, ИПК-5.2, ИПК-5.3 заложены в темах 1-11.

В соответствии с планом на дисциплины студентам выдаются задания на лабораторные работы. Помимо требований и описания функционала в работе указан крайний срок сдачи. Для сдачи лабораторной работы студенту необходимо прислать ссылку на репозиторий в GitHub Classroom и на хостинг, где размещен результат работы с реализованным функционалом, описанным в задании. Работа считается сданной если в ней реализовано 80% и более требований и функционала, описанного в задании.

Каждый студент имеет право на 6 дней опоздания, которые могут быть потрачены на любые задания в течение семестра. Опоздания предназначены для решения особых ситуаций, таких как болезнь или чрезвычайные семейные обстоятельства.

Когда использованы все дни опоздания за каждый день просрочки начисляется штраф в размере 25% от максимального результата за задание. Задания, присланные позже, чем 4 дня, не будут оцениваться. В связи с зависимостью между работами студентам может потребоваться все равно выполнить предыдущие работы, даже если они не оцениваются.

После сдачи лабораторной работы студент должен ее защитить. Во время защиты лабораторной работы преподаватель проверяет репозиторий, хостинг и выполнение критериев и требований задания, а студент отвечает на вопросы преподавателя по его коду, а также теоретических вопросов, приведенных после текста задания лабораторной работы. Если студент отказывается отвечать на вопросы, или дает полностью неверные ответы, или ответы не по теме, то работа может считаться сданной, но при этом она не оценивается.

Работа должна быть выполнена студентом самостоятельно: в репозитории в системе контроля версий студента содержатся коммиты только за его авторством, по этим коммитам можно проследить как велась работа, студент может объяснить свой код и ход выполнения работы, если эти правила не соблюдаются, то работа не считается сданной и не оценивается.

Рубежные контроли пишутся в аудитории индивидуально по варианту задания, выданному преподавателем в назначенные дни. При отсутствии студента в день написания контрольной работы ему дается еще один шанс ее написать на последнем занятии в семестре, но обязательно очно.

Студенты должны заранее сообщать о том, что у них могут возникнуть трудности со своевременной сдачей задания или проекта. При наличии реальных причин задержки студентам следует как можно скорее связаться с преподавателем и обсудить возможные условия.

7.2 Шкала и критерии оценивания результатов обучения

Лабораторная работа оценивается в процентах степени выполнения следующих критериев и для выставления оценки суммируются проценты за каждый из четырех критериев:

1. Полнота выполнения практического задания (30%): соответствует ли функциональность заданным требованиям и целям, насколько точно и без ошибок код выполняет поставленные задачи, насколько эффективно задание отвечает требованиям целевой аудитории и обеспечивает приятное восприятие.

2. Качество и структура кода (10%): качество, читаемость и организация кода, рациональность выполнения задания, последовательность именования и соблюдение лучших практик.

3. Творчество и инновации (10%): творческий подход студентов к выполнению заданий, насколько студенты вышли за рамки основных требований и реализовали дополнительные возможности или использовали уникальные решения.

4. Ответы на вопросы по коду студента и теории (50%):

Дает краткий ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает неправильно (10% из 50%)

Дает развернутый ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает неверно (20% из 50%)

Дает развернутый ответ, содержащий ошибки или неточности. На наводящие вопросы отвечает правильно (30% из 50%)

Дает правильные и развернутые ответы на вопросы (50% из 50%).

R лабораторные рассчитывается как среднее результатов за все лабораторные работы. За полное и безошибочное выполнение всех лабораторных работ в срок и их защиту можно получить максимум 100 баллов (R лабораторные).

Рубежный контроль оценивается по следующим критериям:

Полнота выполнения практического задания: соответствует ли функциональность заданным требованиям и целям, насколько точно и без ошибок код выполняет поставленные задачи.

Качество и структура кода: качество, читаемость и организация кода, рациональность выполнения задания, последовательность именования и соблюдение лучших практик.

Творчество и инновации: творческий подход студентов к выполнению заданий, насколько студенты вышли за рамки основных требований и реализовали дополнительные возможности или использовали уникальные решения.

Пользовательский опыт: отзывчивость, доступность, насколько эффективно задание отвечает требованиям целевой аудитории и обеспечивает приятное восприятие.

Самостоятельность решения: в репозитории студента есть коммиты только за его авторством, по коммитам в репозитории можно проследить как велась работа, студент может

объяснить свой код и ход выполнения работы, если эти правила не соблюдаются, то работа не считается сданной.

Более подробное описание критериев дается в тексте задания рубежного контроля.

За полностью выполненные рубежные контроли также можно получить 100 баллов (R контроль).

Также имеется коэффициент сданных работ K сданные, который равен 1 если все работы сданы и 0 если хотя бы одна работа не сдана.

Итоговый балл рассчитывается по формуле: R сем = (0,5 x R лабораторные + 0,5 x R контроль) * К сданные.

Итоговый балл пересчитывается по шкале ниже и на основании полученной оценки фиксируется результат промежуточной аттестации.

Соответствие баллов в 100 балльной рейтинговой системе оценке по 4-бальной шкале:

0-54 - неудовлетворительно

55-69 - удовлетворительно

70-84 - хорошо

85-100 – отлично

Шкала оценивания	**Описание**
Отлично	Выполнены все **обязательные условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины. Студент демонстрирует соответствие знаний, умений, навыков показателям, приведенным в таблицах, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при переносе знаний и умений на новые, нестандартные задачи. Также требуется выполнение и защита всех лабораторных работ.
Хорошо	Выполнены все **обязательные условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины. Студент демонстрирует соответствие знаний, умений, навыков показателям, приведенным в таблицах, оперирует приобретенными знаниями, умениями, навыками. В то же время при ответе допускает несущественные погрешности, задачи решает с недочетами, не влияющими на общий ход решения. Также требуется выполнение и защита не менее чем девяти лабораторных работ и/или наличие оценки «хорошо» (или выше) по контрольным работам.
Удовлетворительно	Выполнены все **обязательные условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины. Студент демонстрирует соответствие знаний, умений, навыков показателям, приведенным в таблицах, оперирует приобретенными

	знаниями, умениями, навыками. Но показывает неглубокие знания, при ответе не допускает грубых ошибок или противоречий, однако в формулировании ответа отсутствует должная связь между анализом, аргументацией и выводами, в решении задач могут содержаться грубые ошибки. Для получения правильного ответа требуются уточняющие вопросы. Также требуется выполнение и защита не менее чем шести лабораторных работ и/или наличие оценки «удовлетворительно» (или выше) по контрольным работам
Неудовлетворительно	Не выполнены обязательные **условия подготовки студента к промежуточной аттестации**, предусмотренные программой дисциплины, ИЛИ студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями. Выполнено менее 6 лабораторных работ и/или не выполнены контрольные работы.

7.3 Оценочные средства

7.3.1 Текущий контроль
Примерный список вопросов:
Паттерн «Заместитель»
Паттерн «Состояние»
Паттерн «Итератор» и «Компоновщик»
Паттерн «Шаблонный метод»
Паттерны «Адаптер» и «Фасад»
Паттерн «Команда»
Паттерн «Одиночка»
Паттерн «Фабрика»
Паттерн «Декоратор»
Паттерн «Наблюдатель»
Паттерн «Стратегия»
Составные паттерны
Паттерны паттернов
Антипаттерны
«Мост», «Строитель»
«Приспособленец», «Интерпретатор» и «Посредник»
«Хранитель», «Прототип»
Определение MVC
«Посетитель» и «Цепочка обязанностей»

Основные понятия ООП: интерфейс, класс, объект

Основные понятия ООП: метод, сущность, экземпляр

Три основные парадигмы ООП

Нотация UML

Три группы паттернов и их примеры.

Пример задания рубежного контроля

Главные требования

1. Работа выполнена самостоятельно.
2. Работа написана во время занятия.
3. Не менее трех коммитов за авторством студента, равномерно распределенных по времени выполнения задания.
4. Работа размещена на удаленном репозитории на Github, который создан через Github Classroom. (Без этого работа не проверяется)

Задание

1. Принять задание по ссылке на Github Classroom
2. Клонировать репозиторий, который был создан после принятия задания.
3. Финальный вариант когда с историей работы, должен быть в этом репозитории.
4. Проект инициализирован с помощью команды npm init. Имя проекта RK, имя автора - ваше имя.
5. Для запуска сервера есть скрипт с именем start в файле package.json.
6. Проект содержит файл .gitignore с нужными исключениями для node.js.
7. Реализована серверная логика с использованием node.js и express по варианту.
8. Проект структурирован: routes, controllers, services, middlewares, configs.
9. Сервер запускается на порту 5500.
10. Все запросы логируются в консоль. Можно использовать стороннее промежуточное ПО.
11. Есть промежуточное ПО для обработки ошибок, возникающих на стороне сервера и отправляющее сообщению клиенту со статусом 500.
12. Есть промежуточное ПО, которое обрабатывает запросы на несуществующие конечные точки (end points) и отправляет ответ со статусом 400.
13. Есть промежуточное ПО проверяющее наличие заголовка Content-Type и что он содержит application/json, а также на наличие нужных данных в теле запроса согласно варианту, в противном случае отправить ответ {status: "error", message: "Неверные данные"} с кодом 400.
14. Для хранения данных используется субд MongoDB, сервер которой работает на порту 27017.

Важно: Используйте имена свойств в объектах, такие как приведены в варианта, так как они будут использоваться для тестирования на работоспособность ваших работ.