

**Федеральное государственное образовательное бюджетное
учреждение высшего образования**

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ
ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

**Департамент анализа данных и машинного обучения
Факультета информационных технологий и анализа больших данных**

УТВЕРЖДАЮ

**Проректор по учебной и
методической работе**

_____ **Е.А. Каменева**

06.04.2023 г.

Абашин В.Г.

Архитектура и дизайн программного обеспечения

Рабочая программа дисциплины

**для студентов, обучающихся по направлению подготовки
01.03.02 – Прикладная математика и информатика,
ОП «Анализ данных»**

*Рекомендовано Ученым советом
Факультета информационных технологий и анализа больших данных
(протокол №30 от 21.03.2023г.)*

*Одобрено Советом учебно-научного
Департамента анализа данных и машинного обучения
(протокол №1 от 28.02.2023г.)*

Москва 2023

Оглавление

1. Наименование дисциплины3
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине3
3. Место дисциплины в структуре образовательной программы3
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий4
- 5.1. Содержание дисциплины4
- 5.2. Учебно – тематический план6
- 5.3. Содержание семинаров, практических занятий7
6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине8
- 6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы8
- 6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю9
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине10
8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины13
9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины14
10. Методические указания для обучающихся по освоению дисциплины15
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости).16
12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине17

1. Наименование дисциплины

«Архитектура и дизайн программного обеспечения».

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
ПКП-5	Способность применять методы и инструменты анализа данных и машинного обучения при подготовке аналитического обоснования финансово-экономических решений	1. Демонстрирует знание методов и инструментов анализа данных и машинного обучения	Знать: архитектуры современных инструментальных средств реализации моделей машинного обучения Уметь: интегрировать модели машинного обучения
		2. Демонстрирует знание принципов подготовки аналитического обоснования финансово-экономических решений	Знать: основы интеграции компонент моделей машинного обучения Уметь: интегрировать компоненты моделей машинного обучения.
		3. Владеет навыками использования методов и инструментов анализа данных и машинного обучения при подготовке аналитического обоснования финансово-экономических решений	Знать: способы сборки и развертывания многокомпонентных моделей машинного обучения Уметь: собирать и развертывать многокомпонентные модели машинного обучения

3. Место дисциплины в структуре образовательной программы

Дисциплина «Архитектура и дизайн программного обеспечения» относится к Циклу профиля (элективный) по направлению подготовки 01.03.02 - Прикладная математика и информатика, ОП "Анализ данных".

4. Объем дисциплины(модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр 6 (в часах)
Общая трудоемкость дисциплины	3/108	108
<i>Контактная работа – Аудиторные занятия</i>	<i>34</i>	<i>34</i>
<i>Лекции</i>	<i>16</i>	<i>16</i>
<i>Семинары, практические занятия</i>	<i>18</i>	<i>18</i>
Самостоятельная работа	74	74
Вид текущего контроля	Контрольная работа	Контрольная работа
Вид промежуточной аттестации	Зачет	Зачет

5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Архитектуры программных систем

Архитектурные образцы, эталонные модели и эталонные варианты архитектур. Архитектурные структуры и представления. Архитектуры, основанные на уровнях абстракции. Архитектуры, основанные на портах. Архитектуры, основанные на потоках данных. Архитектуры независимых компонентов. Сервис ориентированные архитектуры (SOA).

Тема 2. Архитектурные представления программных систем

Архитектурный вид – структура многослойной программной системы. Архитектурный вид – размещение программной системы. Архитектурный вид – размещение программной системы, основанной на потоках данных. Архитектурный вид – распределение работ по группам разработчиков.

Тема 3. Модульность

Модули, модульно-интерфейсный подход, модельное программирование. Обоснование модульности. Внутренняя характеристика модуля – связанность (прочность). Сцепление модулей – внешняя характеристика модуля.

Тема 4. Сложность программной системы

Методы оценки сложности. Оценка сложности на основе связанности и сцепления модулей.

Тема 5. Представление архитектуры программных систем

Модульно-интерфейсный подход. Объектно-ориентированный подход. Компонентный подход.

Тема 6. Методы структурного проектирования

Методы восходящей разработки. Методы нисходящей разработки. Особенности структурного проектирования.

Тема 7. Формальное описание методики разработки модульной архитектуры программной системы

Проектирование «снизу вверх». Проектирование «снизу вверх». Проектирование архитектуры на основе объектно - ориентированной и компонентной методологии.

Тема 8. Рефакторинг программных систем

Потребность в архитектурном рефакторинге. Построение архитектуры программного средства по ее программному коду. Рефакторинг архитектуры многослойной иерархической программной системы. Возможные подходы к созданию программных средств. Представление созданной архитектуры программного средства. Анализ на соответствие послойной архитектуре (выделение слоев). Коррекция (трансформация) архитектуры в интересах ее рефакторинга. Рефакторинг архитектуры в интересах повышения производительности ИС.

5.2. Учебно – тематический план

№ п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах					Формы текущего контроля успеваемости
		Всего	Контактная работа - Аудиторная работа			Самостоя- тельная работа	
			Общая, в т.ч.:	Лекции	Семинары, практические занятия		
1	Архитектуры программных систем	12	4	2	2	8	Опрос, выполнение индивидуальных заданий контрольная работа
2	Архитектурные представления программных систем	12	6	2	4	6	Опрос, выполнение индивидуальных заданий контрольная работа
3	Модульность	10	4	2	2	6	Опрос, выполнение индивидуальных заданий
4	Сложность программной системы	10	4	2	2	6	Опрос, выполнение индивидуальных заданий
5	Представление архитектуры программных систем	10	4	2	2	6	Опрос, выполнение индивидуальных заданий
6	Методы структурного проектирования	10	4	2	2	6	Опрос, выполнение индивидуальных заданий
7	Формальное описание методики разработки модульной архитектуры программной системы	10	4	2	2	6	Обсуждение, опрос

8	Рефакторинг программных систем	34	4	2	2	30	Обсуждение, опрос
	В целом по дисциплине	108	34	16	18	74	Согласно учебному плану: контрольная работа
	Итого в %		31	47	53	69	

5.3. Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)	Формы проведения занятий
Архитектуры программных систем	Методология решения задач проектирования по Г. Майерсу. Структуры “компонент и соединитель” <i>Рекомендуемые источники:</i> 8.[1],[2],[4]	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений
Архитектурные представления программных систем	Унифицированный язык моделирования UML. Использование базовых диаграмм UML при проектировании ПО : - основные понятия UML, - диаграммы прецедентов, - диаграммы классов, - диаграммы взаимодействий, - диаграммы последовательности действий, - диаграммы состояний, компонентные диаграммы. <i>Рекомендуемые источники:</i> 8.[1],[2],[4]	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений
Модульность	UML. Диаграммы пакетов <i>Рекомендуемые источники:</i> 8.[1],[2],[4]	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений
Сложность программной системы	Методы оценки <i>Рекомендуемые источники:</i> 8.[1],[2],[3]	Интерактивная форма, практикум по решению задач по тематике занятия и

		коллективное обсуждение решений
Представление архитектуры программных систем	UML. Диаграммы компонентов. Диаграммы развертывания <i>Рекомендуемые источники:</i> 8.[1],[2]	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений
Методы структурного проектирования	SADT-модели. DFD-диаграммы <i>Рекомендуемые источники:</i> 8.[1],[5],[6]	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений
Формальное описание методики разработки модульной архитектуры программной системы	Проектирование многослойного программного продукта <i>Рекомендуемые источники:</i> 8.[1],[2],[4]	Занятия в интерактивной форме в виде дискуссий.
Рефакторинг программных систем	Рефакторинг большого класса. Рефакторинг ленивого класса. Способы выявления проблемных участков кода. Методы рефакторинга: улучшения структуры методов в классах, организация данных <i>Рекомендуемые источники:</i> 8.[1],[5],[6]	Занятия в интерактивной форме в виде дискуссий.

6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
--	--	--

Архитектуры программных систем	Структуры «компонент и соединитель». Структуры распределения. Отношения между структурами.	Работа с учебной литературой. Разбор вопросов по теме занятия.
Архитектурные представления программных систем	Архитектурное представление назначения заданий	Работа с учебной литературой. Разбор вопросов по теме занятия. Выполнение и защита домашней контрольной работы
Модульность	Декомпозиционное представление модульной архитектуры и представление зависимостей	Работа с учебной литературой. Разбор вопросов по теме занятия.
Сложность программной системы	Минимизация сложности	Работа с учебной литературой. Разбор вопросов по теме занятия.
Представление архитектуры программных систем	Архитектурное представление назначения заданий	Работа с учебной литературой. Разбор вопросов по теме занятия.
Методы структурного проектирования	Методология решения задач проектирования по Майерсу. Уровни требований к программным системам	Работа с учебной литературой. Разбор вопросов по теме занятия.
Формальное описание методики разработки модульной архитектуры программной системы	Процесс анализа. Процесс синтеза	Работа с учебной литературой. Разбор вопросов по теме занятия.
Рефакторинг программных систем	Рефакторинг в СУБД. Инструментальные средства проведения рефакторинга	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия.

6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Примерный вариант контрольной работы

Построить диаграммы UML для выбранной предметной области:

- диаграммы прецедентов,
- диаграммы классов,
- диаграммы взаимодействий,
- диаграммы последовательности действий,
- диаграммы состояний,

- диаграммы компонентов,
- диаграммы развертывания.

Варианты индивидуального задания

1. Проектирование программы управления ИТ-проектами: задача подбора персонала.
2. Проектирование программы управления ИТ-проектами: задача составления расписания выполнения проекта.
3. Проектирование программы управления ИТ- проектами: задача прогнозирования себестоимости ИТ-проекта.
4. Проектирование программного обеспечения учебного тренажера по выбранной теме.
5. Проектирование программного комплекса для обработки данных на выбранную тематику.
6. Проектирование программного комплекса поддержки принятия управленческих решений (разработка системы учета договоров и расчетов с субподрядчиками)
9. Проектирование программного учёта материально-технических средств.
10. Проектирование клиент- серверного приложения для решения задачи по выбранной тематики.

Критерии бальной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях Департамента анализа данных и машинного обучения.

7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций с указанием индикаторов их достижения в процессе освоения образовательной программы содержится в разделе 2. «Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».

Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, знаний и умений

Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ПКП-5 Способность применять методы и инструменты анализа данных и машинного обучения при подготовке аналитического обоснования финансово-экономических решений	1. Демонстрирует знание методов и инструментов анализа данных и машинного обучения.	Знать: архитектуры современных инструментальных средств реализации моделей машинного обучения Уметь: интегрировать модели машинного обучения	Пример задания Постройте архитектуру системы оценки кредитоспособности (кредитных рисков) лица Пример задания Постройте модель системы оценки кредитоспособности (кредитных рисков) лица
	2. Демонстрирует знание принципов подготовки аналитического обоснования финансово-экономических решений.	Знать: основы интеграции компонент моделей машинного обучения Уметь: интегрировать компоненты моделей машинного обучения.	Пример задания Для заданной предметной области «Обнаружение спама» построить диаграмму пакетов Пример задания Для заданной предметной области «Обнаружение спама» построить диаграмму компонентов
	3. Владеет навыками использования методов и инструментов анализа данных и машинного обучения при подготовке аналитического обоснования финансово-экономических решений	Знать: способы сборки и развертывания многокомпонентных моделей машинного обучения Уметь: собирать и развертывать многокомпонентные модели машинного обучения	Пример задания Осуществите развертывание модели машинного обучения для распознавания образов Пример задания Осуществите развертывание модели машинного обучения ранжирования в информационном поиске

Примерные вопросы для подготовки к зачету

1. Архитектурные образцы, эталонные модели и эталонные варианты

архитектур

2. Архитектурные структуры и представления
3. Архитектуры, основанные на уровнях абстракции
4. Архитектуры, основанные на портах
5. Архитектуры, основанные на потоках данных
6. Архитектуры независимых компонентов
7. Сервис ориентированные архитектуры (SOA)
8. Архитектурный вид – структура многослойной программной системы
9. Архитектурный вид – размещение программной системы
10. Архитектурный вид – размещение программной системы, основанной на

потоках данных

11. Архитектурный вид – распределение работ по группам разработчиков
12. Модули, модульно-интерфейсный подход, модельное программирование
13. Обоснование модульности
14. Внутренняя характеристика модуля – связанность (прочность)
15. Сцепление модулей – внешняя характеристика модуля
16. Методы оценки сложности
17. Оценка сложности на основе связанности и сцепления модулей
18. Модульно-интерфейсный подход
19. Объектно-ориентированный подход
20. Компонентный подход
21. Методы восходящей разработки
22. Методы нисходящей разработки
23. Особенности структурного проектирования
24. Проектирование «снизу вверх»
25. Проектирование «снизу вверх»
26. Проектирование архитектуры на основе объектно - ориентированной и

компонентной методологии

27. Потребность в архитектурном рефакторинге

28. Построение архитектуры программного средства по ее программному коду
29. Рефакторинг архитектуры многослойной иерархической программной системы
30. Возможные подходы к созданию программных средств
31. Представление созданной архитектуры программного средства
32. Анализ на соответствие послойной архитектуре (выделение слоев)
33. Коррекция (трансформация) архитектуры в интересах ее рефакторинга
34. Рефакторинг архитектуры в интересах повышения производительности ИС

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Технология разработки программного обеспечения: учебное пособие / Л. Г. Гагарина, Е. В. Кокорева, Б. Д. Сидорова-Виснадул; под ред. Л. Г. Гагариной. — Москва: ИД «ФОРУМ»: ИНФРА-М, 2023. — 400 с. - ЭБС ZNANIUM.com. – URL: <http://znanium.com/catalog/product/1895679> (дата обращения: 16.02.2023). - Текст : электронный.

2. Назаров, С. В. Архитектура и проектирование программных систем: монография / С. В. Назаров. — 2-е изд., перераб. и доп. — Москва : Инфра-М, 2023. — 374 с. — ЭБС ZNANIUM.com. - URL: <http://znanium.com/catalog/product/1895672> (дата обращения: 16.02.2023). - Текст: электронный.

Дополнительная литература:

3. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем: Учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Электрон. дан. — Москва: Юрайт, 2023. — 432 с. — (Высшее образование). - ЭБС Юрайт. — URL: <https://urait.ru/bcode/513067> (дата обращения: 16.02.2023). — Текст : электронный.

4. Новожилов, О. П. Архитектура ЭВМ и систем в 2 ч. Часть 1 : учебное пособие для академического бакалавриата / О. П. Новожилов. — Москва : Юрайт, 2023. — 276 с.

— (Бакалавр. Академический курс). - ЭБС Юрайт. — URL:<https://urait.ru/bcode/516640> (дата обращения: 16.02.2023). — Текст : электронный.

5. Новожилов, О. П. Архитектура ЭВМ и систем. В 2 ч. Часть 2 : учебное пособие для академического бакалавриата / О. П. Новожилов. — Москва : Юрайт, 2023. — 246 с.— ЭБС Юрайт. — URL: <https://urait.ru/bcode/516641> (дата обращения: 16.02.2023). — Текст : электронный.

6. Галиаскаров, Э. Г. Анализ и проектирование систем с использованием UML : учебное пособие для вузов / Э. Г. Галиаскаров, А. С. Воробьев. — Москва : Юрайт, 2023. — 125 с. — (Высшее образование). — ЭБС Юрайт. — URL: <https://urait.ru/bcode/520341> (дата обращения: 16.02.2023). - Текст : электронный.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Информационно-образовательный портал Финансового университета при Правительстве Российской Федерации <http://portal.ufrf.ru/>
2. Сайт департамента анализа данных и машинного обучения.
3. Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/>
4. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
5. Электронно-библиотечная система «Университетская библиотека ОНЛАЙН» <http://biblioclub.ru/>
6. Электронно-библиотечная система Znanium <http://www.znanium.com>
7. Электронно-библиотечная система издательства «ЮРАЙТ» <https://urait.ru/>
8. Электронно-библиотечная система издательства Проспект <http://ebs.prospekt.org/books>
9. Электронно-библиотечная система издательства «Лань» <https://e.lanbook.com/>
10. Электронная библиотека Издательского дома «Гребенников» <https://grebennikon.ru/>
11. Деловая онлайн-библиотека Alpina Digital <http://lib.alpinadigital.ru/>
12. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>

13. Национальная электронная библиотека <http://нэб.рф/>
14. Финансовая справочная система «Финансовый директор» <http://www.1fd.ru/>
15. Ресурсы информационно-аналитического агентства по финансовым рынкам Cbonds.ru <https://cbonds.ru/>
16. СПАРК <https://spark-interfax.ru/>
17. Academic Reference <http://ar.cnki.net/ACADREF>
18. Bank Focus <http://library.fa.ru/resource.asp?id=527>
19. Информационно-аналитическая база данных EMIS Global <https://www.emis.com/php/companies/overview/index>
20. Реферативная база данных по математике MathSciNET <https://mathscinet.ams.org/mathscinet/>
21. Электронная коллекция книг издательства Springer: Springer eBooks <http://link.springer.com/>
22. Система Thomson Reuters Eikon
23. Web of Science <http://apps.webofknowledge.com>
24. Издательство «Открытые системы» <http://www.osp.ru>

10. Методические указания для обучающихся по освоению дисциплины

Основные этапы работы студента по дисциплине «Архитектура и дизайн программного обеспечения»

1. Предварительная ориентировка в подлежащем изучению учебном материале по программе.
2. Ознакомление с рекомендованной учебной литературой.
3. Слушание и конспектирование лекций, а также выполнение других видов учебной работы.
4. Планирование самостоятельной работы.
5. Обобщение и систематизация информации, почерпнутой из лекций и прочитанной литературы.
6. Выполнение контрольной работы.

Рекомендации по работе с учебным материалом:

1. Осознавайте наличный уровень полученных вами знаний.
2. В ситуации непонимания нужно выявить тот первичный уровень и факторы непонимания, которые стали препятствием понимания последующего.
3. Задавайте сами себе вопросы и пытайтесь ответить на них.

Рекомендации по работе на лекции и с лекционным материалом:

1. Основная задача на лекции – осмысление излагаемого в ней материала. Для этого необходимо слушать лекцию с самого начала, не упуская общих, ориентирующих в материале рассуждений и установок лектора.

2. Ведение записей на лекции важно и полезно для лучшего осмысливания материала, для сохранения информации, с целью ее дальнейшего использования.

3. Для облегчения записи рекомендуется применять сокращения повторяющихся терминов или хорошо известных понятий.

Рекомендации по работе с литературой:

1. Если возникли затруднения при разыскивании материала, по какому-либо конкретному вопросу, следует обратиться к предметному указателю, напечатанному, как правило, в конце каждого литературного источника.

2. Предметный указатель – это алфавитный список основных научных понятий (терминов), содержание которых раскрыто в книге, рядом с термином стоят числа, обозначающие номера страниц, на которых изложен материал, относящийся к данному понятию.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

11.1. Комплект лицензионного программного обеспечения:

Astra Linux, Libre Office

Антивирус Kaspersky

11.2. Современные профессиональные базы данных и информационные справочные системы

Информационно-правовая система «Гарант»

Информационно-правовая система «Консультант Плюс»

Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>

Система комплексного раскрытия информации «СКРИН» -<http://www.skrin.ru/>

11.3. Сертифицированные программные и аппаратные средства защиты информации - не используются

11.4. Python 3.

11.5. StarUML.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционные занятия проводятся в мультимедийных аудиториях, а семинарские занятия – в компьютерных классах.