

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ**

Кафедра автоматизации обработки информации (АОИ)

М. М. Милихин

ПРОЕКТИРОВАНИЕ И АРХИТЕКТУРА ПРОГРАММНЫХ СРЕДСТВ

**Методические указания
по выполнению лабораторных работ для студентов ФДО
направления подготовки 09.03.04 «Программная инженерия»**

Томск 2017

Корректор: А. Н. Миронова

Милихин М. М.

Проектирование и архитектура программных средств : методические указания по выполнению лабораторных работ для студентов ФДО направления подготовки 09.03.04 «Программная инженерия» / М. М. Милихин. – Томск : ФДО, ТУСУР, 2017. – 14 с.

Методические указания содержат описание методики проведения лабораторных работ и рекомендации по их выполнению.

© Милихин М. М., 2017
© Оформление.
ФДО, ТУСУР, 2017

СОДЕРЖАНИЕ

Основные сокращения и условные обозначения	4
1 Введение	5
2 Основные требования к оформлению отчета	8
3 Лабораторная работа № 1 «Моделирование поведения программных средств на логическом уровне»	9
3.1 Цель работы	9
3.2 Порядок выполнения работы	9
3.3 Вопросы для самостоятельной подготовки	11
4 Лабораторная работа № 2 «Проектирование структуры программного продукта»	12
4.1 Цель работы	12
4.2 Порядок выполнения работы	12
4.3 Вопросы для самостоятельной подготовки	13
Приложение А Индивидуальные варианты заданий	14

ОСНОВНЫЕ СОКРАЩЕНИЯ И УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

ООАП – объектно-ориентированный анализ и проектирование

ПО – программное обеспечение

ПП – программный продукт

ПС – программное средство

UML – унифицированный язык моделирования

1 ВВЕДЕНИЕ

В рамках дисциплины «Проектирование и архитектура программных систем» изучаются процесс проектирования программных систем, методы управления проектом по созданию программных систем, технологии разработки интегрированных моделей программных систем с использованием графических инструментальных сред проектирования. Дисциплина нацелена на формирование у студента навыков сбора, обработки и представления исходных данных для принятия проектных решений; моделирования архитектуры программных систем; объектно-ориентированного анализа и проектирования.

Дисциплина относится к базовой части профессионального цикла дисциплин блока 1 «Дисциплины (модули)».

Для эффективного освоения дисциплины студент должен знать основные положения дисциплин: «Введение в программную инженерию», «Разработка и анализ требований», «Объектно-ориентированный анализ и программирование».

Знания и навыки, полученные в процессе освоения дисциплины, будут востребованы при дальнейшем изучении таких дисциплин учебного плана по данному направлению подготовки, как «Тестирование программного обеспечения», «Хранилища данных», «Подготовка и защита выпускной квалификационной работы», «Преддипломная практика».

Целью проведения лабораторных работ по дисциплине является закрепление теоретических знаний и получение практических навыков проектирования программных средств.

При организации и проведении лабораторных работ решаются следующие задачи: освоение графических инструментальных сред проектирования архитектуры и структуры программного обеспечения, получение практических навыков построения диаграмм в нотации унифицированного

языка моделирования (UML), практическое применение полученных в рамках изучения дисциплины знаний об архитектуре и архитектурных стилях и шаблонах современного программного обеспечения.

Методические указания включают описание двух лабораторных работ, в которых решаются задачи построения нескольких взаимосвязанных канонических диаграмм UML для выбранного программного продукта.

В результате выполнения лабораторных работ обучающийся должен закрепить знания языка UML, освоить основные принципы разработки структуры и архитектуры программных продуктов, развить навыки проектирования сложных программных систем.

Выполнение лабораторных работ направлено на формирование компетенций ОПК-3 (готовность применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов) и ПК-3 (владение навыками использования различных технологий разработки программного обеспечения); позволит получить практические навыки проектирования программных средств и облегчит работу над курсовым проектом.

Лабораторные работы выполняются в соответствии с порядком, описанным в методических указаниях.

Выбор варианта лабораторной работы осуществляется по общим правилам с использованием следующей формулы:

$$V = (N \times K) \text{ div } 100,$$

где V – искомый номер варианта,

N – общее количество вариантов,

div – целочисленное деление,

при $V = 0$ выбирается максимальный вариант,

K – код варианта.

В качестве инструмента для разработки диаграмм UML студент может использовать любой редактор UML-диаграмм, например **Dia** (распространяется свободно, доступен для скачивания на официальном сайте проекта <http://dia-installer.de>, Microsoft Visio, IBM RationalRose).

Автор учебно-методического пособия предлагает использовать сервис **draw.io**, для доступа к которому необходимо перейти по адресу <https://www.draw.io/> в любом современном графическом интернет-браузере с поддержкой технологий HTML5. Инструменты разработки диаграмм UML доступны в разделе UML панели инструментов.

2 ОСНОВНЫЕ ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТА

Для успешного выполнения лабораторной работы необходимо предварительно изучить цель работы и основные теоретические сведения в рамках темы работы.

Результатом выполнения лабораторной работы является отчет, оформленный в соответствии с требованиями, представленными в методических указаниях.

Отчет о выполнении лабораторной работы должен содержать наименование темы лабораторной работы, цель и задачу исследования, а также детальное описание результатов выполнения каждого этапа работы.

При оформлении отчетов по текстовым работам (контрольным и лабораторным) следует руководствоваться требованиями образовательного стандарта вуза: ОС ТУСУР 01–2013. Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления. – Режим доступа: <https://regulations.tusur.ru/documents/70>

3 ЛАБОРАТОРНАЯ РАБОТА № 1

«МОДЕЛИРОВАНИЕ ПОВЕДЕНИЯ ПРОГРАММНЫХ СРЕДСТВ НА ЛОГИЧЕСКОМ УРОВНЕ»

3.1 Цель работы

Целью работы является приобретение навыков моделирования поведения компонентов программных средств с использованием канонических диаграмм состояний и деятельности языка UML.

Решаемые в ходе выполнения работы задачи:

- построение диаграммы состояний выбранных объектов в рамках определенного программного продукта;
- построение диаграммы деятельности для выбранных алгоритмов или методик, реализуемых в рамках выбранного программного продукта.

3.2 Порядок выполнения работы

Задание на лабораторную работу включает в себя последовательное выполнение следующих действий.

1. Ознакомиться с теоретическим материалом (п. 5.2 и 5.3 учебного пособия¹).
2. Выбрать предметную область для проектирования из предложенного списка вариантов заданий (приложение А), согласно варианту, рассчитанному по формуле.
3. Сформулировать краткое текстовое описание выбранной предметной области (несколько предложений).
4. Выбрать два объекта в рамках выбранной предметной области (в контексте диаграммы состояний UML), для которых характерны нали-

¹ Милихин М. М. Проектирование и архитектура программных средств : учеб. пособие / М. М. Милихин, М. М. Рычагов. – Томск : ФДО, ТУСУР, 2015. – 138 с.

чие нескольких (не менее пяти) состояний на разных стадиях работы разрабатываемого программного продукта (ПП) и переходов между ними.

5. Сформулировать текстовое описание объектов и их возможных состояний.

6. Разработать диаграмму состояний для выбранных объектов в нотации UML.

7. Выбрать один из наименее тривиальных алгоритмов, которые необходимо реализовать в рамках выбранной предметной области.

8. Сформулировать текстовое пояснение выбранного алгоритма и его основных особенностей.

9. Разработать диаграмму деятельности для выбранного алгоритма.

Отчет о выполнении лабораторной работы должен содержать следующие сведения:

- вариант индивидуального задания и краткое описание выбранной предметной области;
- результат выполнения каждого пункта задания на лабораторную работу, включая текстовые пояснения полученных результатов и графические нотации следующих диаграмм:

- диаграммы состояний двух выбранных объектов, а также текстовые пояснения к ним в соответствии с заданием на лабораторную работу;

- диаграммы деятельности выбранного алгоритма, а также текстовые пояснения к ним в соответствии с заданием на лабораторную работу.

Трудоемкость выполнения задания составляет 5 часов.

3.3 Вопросы для самостоятельной подготовки

1. В чем заключается отличие термина «объект» в контексте диаграммы состояний UML от общепринятого значения в ООАП?
2. Для чего используется диаграмма состояний UML?
3. В чем заключается отличие понятия состояния на диаграммах состояний и деятельности?
4. Сформулируйте основные отличия диаграммы деятельности UML от традиционной блок-схемы.
5. Перечислите основные элементы диаграммы состояний UML.
6. Перечислите основные элементы диаграммы деятельности UML.
7. Сформулируйте основные понятия, входящие в формализм автомата.
8. Перечислите особенности начального и конечного состояний диаграммы состояний UML.
9. Как можно обозначить параллельное выполнение операций на диаграмме деятельности?
10. Сформулируйте различие между состоянием действия и состоянием деятельности.

4 ЛАБОРАТОРНАЯ РАБОТА № 2

«ПРОЕКТИРОВАНИЕ СТРУКТУРЫ ПРОГРАММНОГО ПРОДУКТА»

4.1 Цель работы

Целью настоящей работы является приобретение навыков разработки структуры программных средств в соответствии с выбранным архитектурным шаблоном с использованием канонической диаграммы классов UML.

Решаемая в ходе выполнения работы задача – построение диаграммы классов проектируемого программного продукта.

4.2 Порядок выполнения работы

Индивидуальный вариант задания соответствует теме, выбранной для выполнения первой лабораторной работы. Задание на лабораторную работу включает в себя последовательное выполнение следующих действий.

1. Ознакомиться с теоретическим материалом (п. 5.4 и гл. 6 учебного пособия²).

2. Перечислить основные функциональные и нефункциональные требования к разрабатываемому программному обеспечению.

3. Выбрать архитектурные стили, наиболее подходящие для реализации перечисленных требований.

4. Построить диаграмму классов разрабатываемого ПП.

5. Сформулировать текстовое описание каждого класса, представленного на диаграмме классов, и его основных методов.

6. Описать влияние выбранных архитектурных стилей на структуру классов разрабатываемого ПП.

² Милихин М. М. Проектирование и архитектура программных средств : учеб. пособие / М. М. Милихин, М. М. Рычагов. – Томск : ФДО, ТУСУР, 2015. – 138 с.

Отчет о выполнении лабораторной работы должен содержать следующие сведения:

- вариант индивидуального задания и краткое описание выбранной предметной области;
- графическую нотацию диаграммы классов и текстовое описание для каждого ее элемента;
- текстовые пояснения к результатам выполнения каждого этапа лабораторной работы.

Трудоемкость выполнения задания составляет 5 часов.

4.3 Вопросы для самостоятельной подготовки

1. Для чего в ООАП используется диаграмма классов?
2. Что такое класс?
3. Какие основные элементы описания класса Вы знаете?
4. Отношения каких типов могут быть использованы на диаграмме классов?
5. Что такое объект в терминологии ООАП?
6. Какие архитектурные стили Вам известны?
7. Зачем нужно использовать архитектурные стили при проектировании архитектуры ПО?
8. В чем разница между архитектурой и структурой ПО?
9. Что такое квантор видимости?
10. Какие требования предъявляются к наименованию классов?

ПРИЛОЖЕНИЕ А

Индивидуальные варианты заданий

1. Городская справочная геоинформационная система.
2. Корпоративный сайт компании, занимающейся разработкой программного обеспечения.
3. Файлообменный сервис.
4. Интернет-магазин электроинструментов.
5. Сервис онлайн-записи на тест-драйв автомобиля.
6. Приложение для смартфонов «Русско-английский разговорник».
7. Графический текстовый редактор.
8. Видеопроектор для персонального компьютера.
9. Сетевая игра «Шахматы».
10. Программная система «Мобильный банк».
11. Сайт фитнес-клуба.
12. Система организации видеоконференций.
13. Электронное расписание высшего учебного заведения.
14. Система «умный дом».
15. Облачная геоинформационная система.
16. Интернет-сервис по созданию презентационной графики.
17. Кроссплатформенный аудиоплеер.
18. Система обмена текстовыми и мультимедийными сообщениями.
19. Конвертер величин.
20. Кулинарная книга с поиском рецептов по заданному набору продуктов.