



ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
УПРАВЛЕНИЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ И ПОВЫШЕНИЯ
КВАЛИФИКАЦИИ

Кафедра «Программное обеспечение вычислительной тех-
ники и автоматизированных систем»

Учебно-методическое пособие
по дисциплине

**«ПРОЕКТИРОВАНИЕ И
АРХИТЕКТУРА
СЛОЖНЫХ ПРОГРАММНЫХ
СИСТЕМ»**

Автор
Долгов В.В.

Ростов-на-Дону, 2018

Аннотация

Учебно-методическое пособие предназначено для студентов очной формы обучения направления 09.04.04 «Программная инженерия».

Авторы

доцент, к.т.н.,
зав.каф. ПОВТиАС
Долгов В.В.



Оглавление

1. Общие указания к выполнению лабораторных работ	4
1.1. Требование к лабораторному оборудованию	4
1.2. Требования, предъявляемые при сдаче лабораторных работ.....	4
2. Лабораторная работа №1: Разработка многослойного приложения работы с данными	5
2.1. Цель работы	5
2.2. Задание к лабораторной работе	5
2.3. Контрольные вопросы.....	7
3. Лабораторная работа №2: Установка и работа с очередями сообщений	7
3.1. Цель работы	7
3.2. Задание к лабораторной работе	7
3.3. Контрольные вопросы.....	8
Список литературы	9

1. Общие указания к выполнению лабораторных работ

1.1. Требование к лабораторному оборудованию

Аудитория для проведения лабораторных занятий должна быть укомплектована специализированной мебелью и компьютерами следующей минимальной комплектации:

- Процессор: не менее двух исполнительных ядер, совместимый с системой команд x86 и x64, с поддержкой аппаратной виртуализации.
- Оперативная память: не менее 4 Гб.
- Монитор: не менее 22" (дюймов) по диагонали.
- Наличие сети не менее 100 Мб/сек.
- Наличие клавиатуры и манипулятора «мышь».

На компьютерах должно быть установлено следующее программное обеспечение:

- Операционная система: Microsoft Windows 7 (или выше).
- Одно или несколько специализированных сред разработки программного обеспечения: Microsoft Visual Studio 2012 (или выше), IntelliJ IDEA, Eclipse, Microsoft Visio (или полный аналог).
- Компьютер должен быть обеспечен доступом в глобальную сеть Интернет на скорости не менее 1 Мбит/сек.

1.2. Требования, предъявляемые при сдаче лабораторных работ

При сдаче студентом лабораторной работы отчетом выступает исходный код созданной в процессе выполнения работы программы.

Исходный код должен быть отформатирован согласно принятым для языка F# стандартам. Допускается наличие в исходном коде комментариев, описывающих основные части программы и особенности их функционирования. В то же время студент должен быть готов объяснить работу программы в целом и каждую отдельную ее часть при полном отсутствии комментариев (например, они могут быть удалены или сделаны нечитаемыми перед сдачей).

Обязательным условием сдачи является умение студента восстановить любой участок исходного кода программы после его удаления (при этом этот участок должен быть набран студентом заново, запрещается использование операции Undo текстовых редакторов).

Исходный код сдаваемой программы должен быть пред-

ставлен в электронном виде.

Сдача исходного кода ранее сдававшихся программ или программ, код которых выложен в сети Интернет, не допускается.

2. ЛАБОРАТОРНАЯ РАБОТА №1: РАЗРАБОТКА МНОГОСЛОЙНОГО ПРИЛОЖЕНИЯ РАБОТЫ С ДАННЫМИ

2.1. Цель работы

Закрепить навыки разработки приложений с разделением внутреннего функционала на различные по своему назначению архитектурные слои. Интеграция разработанных в ходе выполнения практических заданий элементов проекта в единое программное средство.

2.2. Задание к лабораторной работе

В соответствие с вариантом задания, разработайте программное средство, выполняющее указанный функционал. Внутренняя структура программного средства должна состоять из трех слоев: слой доступа к данным, слой бизнес-логики, слой представления данных (интерфейс). Проектирование и реализация каждого слоя должна вестись в рамках соответствующего практического занятия. Время, отводимое на выполнение лабораторной работы, должно использоваться для стыковки различных слоев, отладки и сдачи программного средства.

Тексты заданий по вариантам приведены в таблице 1.

Таблица 1 – Варианты заданий к лабораторной работе №1

№ варианта	Задание
------------	---------

1	<p>Разработать программное средство, позволяющее вести учет успеваемости студентов высшего учебного заведения. Программное средство должно минимально включать следующие данные:</p> <ul style="list-style-type: none"> – информацию о студентах, обучаемых в ВУЗе; – информацию о читаемых дисциплинах; – информацию о принадлежности студента к той или иной группе; – информацию о преподавателях и о дисциплинах, закрепленных за ними (учтите, что набор закрепленных дисциплин может меняться от года к году); – информацию об оценках, полученных студентами по результатам изучения той или иной дисциплины (а также о том, кто из преподавателей ее поставил).
2	<p>Разработать программное средство, позволяющее вести учет научных статей, написанных коллективами авторов и изданных в научных журналах. Программное средство должно минимально позволять оперировать следующими данными:</p> <ul style="list-style-type: none"> – списком авторов (с указанием места их работы и должности); – список научных журналов (с указанием названия, периодичности выхода, индекса цитируемости, редакционной коллегии, списка вышедших номеров); – списком статей (с указанием списка авторов, в каком номере и какого журнала она опубликована, какие статьи ссылаются на данную, индекса цитируемости).

3	<p>Разработать программное обеспечение учета коммерческих лицензий на программное обеспечение. Программное средство в минимуме должно обладать возможностью учета следующих данных:</p> <ul style="list-style-type: none"> – перечень помещений/комнат/аудиторий коммерческой фирмы (с указанием номера, этажа, ответственного); – парк компьютерной техники с разделением на сервера и рабочие станции (краткая аппаратная конфигурация, расположение); – перечень лицензий на программное обеспечение (тип лицензии, способ лицензирования, текст лицензионного соглашения, количество лицензий, срок действия лицензии); – перечень лицензий, установленных на тот или иной компьютер.
---	---

2.3. Контрольные вопросы

1. В чем причины появления многоуровневой архитектуры приложений?
2. Всегда ли удастся изолировать слой представления от слоя данных (какова цена такой изоляции)?
3. Приведите критерии, по которым происходит отнесение кода программы к слою данных, либо к слою бизнес-логики.
4. Все ли проверки целостности данных можно выполнить в слое доступа к данным?
5. Какие подходы к организации слоя представления вы можете назвать?
6. Перечислите способы реализации слоя данных. Опишите плюсы и минусы каждого из подходов.

3. ЛАБОРАТОРНАЯ РАБОТА №2: УСТАНОВКА И РАБОТА С ОЧЕРЕДЯМИ СООБЩЕНИЙ

3.1. Цель работы

Получить практические навыки установки, настройки и использования в собственном программном продукте одной из систем обмена сообщениями. Получить навыки разработки приложений как совокупности сетевых сервисов.

3.2. Задание к лабораторной работе

Проектирование и архитектура сложных программных систем

1. Выберите и установите одну из систем обмена сообщениями (ActiveMQ, RabbitMQ, Apache Kafka, Azure Service Bus). Установка может происходить как на физические компьютеры, так и на виртуальные серверы (в том числе с использованием Docker-контейнеров).

2. Используя результаты выполнения практического занятия №4 «Проектирование и разработка контрактов для сервисной модели доступа», выполните декомпозицию проекта (в соответствии с заданием из лабораторной работы №1) с реализацией его частей в виде отдельных сетевых служб.

3.3. Контрольные вопросы

1. Какие преимущества привносят очереди сообщений в архитектуру сложных приложений? Какие недостатки они имеют?

2. Обоснуйте свой выбор той или иной очереди сообщений для реализации проекта.

3. Каким образом очереди сообщений влияют на связность проекта?

4. Назовите основные признаки служб, которые могут быть реализованы в виде сервисов.

СПИСОК ЛИТЕРАТУРЫ

1. Сошников Д.В. Функциональное программирование на F#. – ДМК Пресс, 2011.
2. Robert Pickering, Foundations of F#. – Apress, 2007.
3. Chris Smith, Programming F#. – O'Reilly, 2010.
4. Березина Н.И. Лабораторные работы по курсу «Объектно-ориентированное программирование: язык программирования C#» : Учебное пособие – М.: Издательский отдел факультета ВМиК МГУ имени М.В.Ломоносова, 2010.