



**Федеральное государственное образовательное бюджетное учреждение
высшего образования**

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

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

Д.М. Милованов, И.Я Хасаншин

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ
КУРСОВОЙ РАБОТЫ ПО ДИСЦИПЛИНЕ «СОВРЕМЕННЫЕ
ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ»**

4 семестр

Для студентов,
обучающихся по направлению
09.03.03 «Прикладная информатика»,
(программа подготовки бакалавра)

Москва 2021

**Федеральное государственное образовательное бюджетное учреждение
высшего образования**

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

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

Д.М. Милованов, И.Я. Хасаншин

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ
КУРСОВОЙ РАБОТЫ ПО ДИСЦИПЛИНЕ «СОВРЕМЕННЫЕ
ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ»**

4 семестр

Для студентов,
обучающихся по направлению
09.03.03 «Прикладная информатика»,
(программа подготовки бакалавра)

*Одобрено Советом Департамента анализа данных и машинного обучения
(протокол №7 от 27.01.2021 г.)*

Москва – 2021

УДК 004(072)
ББК 32.973
М 60

Авторы:

Милованов Д.М., ассистент Департамента анализа данных и машинного обучения факультета информационных технологий и анализа больших данных Финансового университета при Правительстве РФ

Хасаншин И.Я., канд. тех. наук, доцент Департамента анализа данных и машинного обучения факультета информационных технологий и анализа больших данных Финансового университета при Правительстве РФ

Рецензенты:

Коротеев М.В., канд. эк. наук, доцент Департамента анализа данных и машинного обучения факультета информационных технологий и анализа больших данных Финансового университета при Правительстве РФ

М 60 Милованов Д.М., Хасаншин И.Я.

Методические материалы по выполнению курсовой работы по дисциплине «Современные технологии программирования», 4 семестр. Для студентов, обучающихся по направлению 09.03.03 «Прикладная информатика», (программа подготовки бакалавра) – М.: Финансовый университет, департамент анализа данных и машинного обучения, 2021. – 16 с.

Цель методических указаний – предоставление необходимого методического обеспечения по выполнению, оформлению и защите курсовой работы по дисциплине «Современные технологии программирования». Методические указания содержат темы курсовых работ, требования к выполнению, систему оценивания, методические рекомендации студенту для выполнения и подготовки к сдаче курсовой работы.

Методические указания предназначены для бакалавров направления подготовки 09.03.03 «Прикладная информатика».

УДК 004(072)
ББК 32.973

Учебное издание

*Милованов Даниил Михайлович,
Хасаншин Ильшат Ядыкарович*

**Методические материалы по выполнению курсовой работы по дисциплине
«Современные технологии программирования»**

Для студентов, обучающихся по направлению 09.03.03 «Прикладная информатика»,
(программа подготовки бакалавра)

Компьютерный набор, верстка Д.М. Милованов

Электронное издание

© ФГОБУ ВО «Финансовый университет при
Правительстве Российской Федерации», 2021
© Милованов Даниил Михайлович, 2021.
© Хасаншин Ильшат Ядыкарович, 2021.

Содержание

I.	5	
II.	5	
III.	6	
IV.	7	
V.	9	
VI.	12	
		Рекомендации по разработке программы
		8
		Рекомендации по пользовательскому интерфейсу.
		9
		Рекомендации по защите работы
		10
VII.	14	
VIII.	15	

I. Общее задание на курсовую работу

Требуется разработать кросс-платформенное приложение (Клиент) и приложение, которое занимается хранением и обработкой информации, (Сервер), выполняющее учет и отображение объектов предметной области, хранящихся в базе данных (БД), и предоставляющее справочную и статистическую информацию. Информационная модель предметной области должна быть не проще двухуровневой структуры данных типа «родитель-дочка».

Клиент должен быть с графическим интерфейсом. Реализация допускается на языке программирования Java (графический интерфейс реализуется при помощи Swing или JavaFX). Клиент должен уметь соединяться с сервером при помощи запросов к API Сервера.

Сервер может не иметь интерфейса. Задача сервера, обрабатывать данные, полученные от Клиента, сохранять их в БД, делать запросы к БД. Сервер должен иметь реализацию RESTful API для взаимодействия с клиентом.

II. Общие требования к курсовой работе

Общие требования:

1. В курсовом проекте должна быть разработана информационная модель предметной области, представленная в виде пользовательских классов и таблиц БД. Взаимодействие с БД должно быть реализовано при помощи ORM.

2. Должно быть разработано несколько форм пользовательского интерфейса для клиента.

3. Разработчик самостоятельно определяет интерфейс программы и ее функциональность, однако для получения максимальной оценки приложение

в обязательном порядке независимо от предметной области, указанной в задании, должно выполнять следующие операции:

- Отображать в таблице данные предметной области:
- Для информационной модели, основанной на БД, таблицы должны быть предварительно заполнены записями.
- Реализовать добавление в БД нового объекта, удаление объекта, редактирование объекта.
- Реализовать фильтрацию записей БД, удовлетворяющих введенному пользователем сложному критерию.
- Реализовать сортировку записей.
- Обновлять изменения источника данных в базе данных.
- После сохранения данных при запуске программы загрузить данные из БД.
- Создать пункт меню «Об авторе».
- Разработать несколько полезных пользователю функций для отображения статистических данных, например, средних, максимальных или минимальных значений, данных для построения гистограммы или графика и т.п.

4. Программа не должна завершаться аварийно: сообщения о некорректном вводе данных, противоречивых или недопустимых значениях данных, при отсутствии данных по функциональному запросу пользователя и других нештатных ситуациях отображать в окнах сообщений.

5. Программа должна иметь содержательные комментарии, которые могут генерировать автоматически составляемую документацию при помощи инструмента JavaDoc.

III. Структура пояснительной записки.

Титульный лист

Оглавление

Введение

1. Постановка задачи
2. Описание предметной области
3. Актуальность автоматизации
4. Описание программы
 - 4.1. Алгоритмические решения
 - 4.2. Описание интерфейса программы
 - 4.3. Состав приложения
5. Назначение и состав классов программы

Заключение

Список литературы

Приложение. 1. Исходный код программы

IV. Порядок и сроки выполнения курсовой работы

1. Получение от ответственного за КР список тем с фамилиями руководителей.
2. Выбор темы согласовывается со старостой – на одну тему – один исполнитель в группе. Староста передает ответственному за КР список группы.
3. Ответственный передает список тем КР на подпись. Изменить тему после утверждения нельзя.
4. После утверждения тем, на информационно-образовательном портале (ИОП) университета каждому студенту станет доступна таблица с полями: тема, результат проверки на плагиат, загрузка работы, отзыв, допуск к защите, оценка и др.

5. Студент разрабатывает программу, пояснительную записку и презентацию.
6. Пояснительная записка должна быть сброшюрована по правилам.
7. Показ работ руководителю, исправление ошибок и получение разрешение на загрузку работы на ИОП.
8. Загрузка работы (Пояснительной записки) и результаты проверки работы на плагиат на портал.
9. Руководитель формирует отзыв и загружает его в таблицу.
10. Сдача отзыва и работы в распечатанном виде специальному ответственному за КР от департамента.
11. Получение допуска к защите. Назначается место, дата и время защиты.
12. Происходит защита работ в назначенное дату и время.
13. Руководитель выставляет оценку.

Рекомендуемые контрольные сроки: в конце **апреля** работа должна быть готова к защите.

Сроки подготовки и защиты курсовых работ (см. на портале fa.ru)

Выбор темы курсовой работы студентами	до 19 февраля
Закрепление темы курсовой работы за студентом и назначение руководителя	до 25 февраля
Утверждение плана курсовой работы руководителем	до 05 марта
Представление выполненной курсовой работы в департамент	до 05 мая
Защита курсовых работ	до 26 мая

V. Показатели оценивания курсовой работы

Оценка за курсовую работу (О) складывается из:

- оценки за качество и полноту реализации приложения (Опр);
- оценки за пояснительную записку (Опз);
- оценки за презентацию и защиту КП (Оз).

Итоговая оценка вычисляется по формуле:

$$O=0,6 \text{ Опр} + 0,25 \text{ Опз}+0,15 \text{ Оз}.$$

Показатели оценивания	Макс. балл
1. Качество и полнота реализации приложения	
<p>Требования на оценку «отлично»</p> <ol style="list-style-type: none"> 1. Программа полностью соответствует заданию. 2. Программа реализована в трехзвенной архитектуре (слой клиента – слой сервера – слой данных) 3. Программа ни при каких обстоятельствах не завершается аварийно. 4. Пользовательский интерфейс эргономичен и интуитивно понятен. 5. Алгоритмы обработки данных эффективны и рациональны. 6. Программа соответствует требованиям объектно-ориентированного и структурного программирования. 7. Программа читабельна и задокументирована. 8. Программа задокументирована по стандарту JavaDoc 	86-100

<p>9. Обнаружены отдельные недостатки по п. 1-6, но программа имеет повышенную сложность и/или большой объем.</p>	
<p>Требования на оценку «хорошо»</p> <p>Не выполнены полностью требования на оценку «отлично» по одному или нескольким критериям, при этом выявлены один или несколько недостатков:</p> <ol style="list-style-type: none"> 1. Программа в основном соответствует заданию. Допущены несущественные отклонения от условия. 2. Программа реализована в двухзвенной (слой клиента – слой базы данных) или трехзвенной архитектуре 3. Программа завершается аварийно только при вводе некорректных данных или выполнении второстепенных функций. 4. Пользовательский интерфейс недостаточно эргономичен. 5. Существуют более эффективные алгоритмы обработки данных и/или более простой способ решения задачи. 6. Обнаружены отдельные недоработки. 7. Обнаружены отдельные недостатки по п. 1-5, но программа имеет повышенную сложность и/или большой объем. 	70-85
<p>Требования на оценку «удовлетворительно»</p> <p>Не выполнены требования на оценку «хорошо» по одному или нескольким критериям, но программа может быть выполнена и при этом будут выявлены дополнительные недостатки:</p> <ol style="list-style-type: none"> 1. Задание выполнено частично. 	50-69

<p>2. Программа реализована в двухзвенной (слой клиента – слой базы данных) или трехзвенной архитектуре</p> <p>3. Программа написана некачественно, содержит ошибки.</p> <p>4. Программа аварийно завершается при выполнении некоторых функций.</p>	
<p>Оценка «неудовлетворительно» выставляется в остальных случаях:</p> <ol style="list-style-type: none"> 1. В целом задание не выполнено. 2. Отсутствует выполняемая программа. 3. Отсутствует проект. <p>Курсовая работа не допускается к защите</p>	0-49
<p>II. Качество пояснительной записки</p>	
<ol style="list-style-type: none"> 1. Соответствие оглавления пояснительной записки требованиям. 2. Полнота раскрытия вопросов. 3. Соответствие текста пояснительной записки предъявляемым требованиям к оформлению. 4. Наличие аргументированной точки зрения и обоснований принятия решений. 5. Аккуратность оформления, отсутствие орфографических и стилистических ошибок. 6. Соответствие схем ГОСТу. 	0-100
<p>III. Защита курсовой работы</p>	
<ol style="list-style-type: none"> 1. Качество доклада. 2. Качество презентации. 3. Ответы на вопросы. 4. Сроки представления 	0-100

VI. Рекомендации по выполнению курсовой работы

Примечание. Рекомендации носят необязательный характер, однако их необоснованное игнорирование может привести к снижению оценки за КР.

Рекомендации по разработке программы

Программа ни при каких условиях не должна аварийно завершаться или выдавать неверный результат из-за ошибок пользователя! Везде, где это необходимо, использовать обработку исключений и проверки. Проверять преобразование исходных данных.

Алгоритмы должны быть эффективными и простыми. Программа должна быть читабельной и содержать комментарии. В идеале соответствовать концепции «Чистый код».

Должна быть реализована концепция ООП

Запоминать, если необходимо, параметры программы в ресурсах, а при запуске программы автоматически считывать файл и производить ее настройку на эти параметры. В качестве параметров могут быть запомнены: имя файла объектов, в котором при выходе из программы была сохранена информация; состояние переключателей и флажков, настройки окна, и т.д. При запуске программы автоматически открывать запомненный файл.

Реализовать работу с программой пользователей разного уровня (администратор, пользователь). Если в программе необходимо вводить логины и пароли, то они должны быть отображены рядом с полями ввода. В БД они должны храниться в хешированном виде.

Рекомендации по пользовательскому интерфейсу.

Пользовательский интерфейс должен быть эргономичным, интуитивно понятным и эстетичным. Цветовая палитра формы должна соответствовать хорошему стилю.

Если в столбце таблицы отображаются фиксированные данные, например названия факультетов, то столбец должен быть преобразован в виде элемента выбора comboBox.

Программа должна отображать не менее 4-х взаимосвязанных по данным окон. Окна не должны быть перегружены элементами управления: редко используемые возможности необходимо концентрировать в других окнах или вкладках.

Интерфейс должен быть минимизирован: если что-то можно сделать автоматически или за одно действие вместо нескольких, надо такой интерфейс и реализовывать. Например, вместо использования кнопок, можно автоматически отображать количественные данные в строке состояния.

В формах рекомендуется использовать элементы управления: меню, панель инструментов, кнопки, надписи, сетку, подсказки, полосу состояния и т.д.

Размещение элементов управления должно отвечать требованиям эргономики.

Элементы управления, которые в данный момент не могут использоваться, должны быть либо неактивны, либо невидимы.

Известные значения должны быть представлены счетчиками, переключателями и списками, допускающими добавление и удаление элементов. К известным значениям могут быть отнесены: текущая дата; наиболее предполагаемое значение; списки фамилий, названий и т.д.. Если эти значения не являются заранее известными, то их можно динамически выделить из хранилища объектов.

Рекомендации по защите работы

После допуска студента к защите, защита выполняется в дни и часы, назначенные руководителем. Студент выполняет доклад по выполненной

работе. Время, отводимое на доклад: 10 минут. В докладе необходимо отразить постановку задачи с данными о разработчике, иллюстрацию предметной области, диаграмму или схему информационной модели предметной области, основные окна программы, примеры выполнения основных функций.

К докладу должно быть подготовлено 7-10 слайдов в формате PowerPoint по освещаемым вопросам доклада. Основным содержанием слайдов должны быть схемы, рисунки, таблицы, графики. Отображать на слайде текст доклада не рекомендуется

VII. Список примерных тем курсовых работ

1. Информационно-справочная система туристического агентства
2. Информационно-справочная система жд вокзала
3. Информационно-справочная система аэропорта
4. Информационно-справочная система ресторана
5. Информационно-справочная система библиотеки
6. Информационно-справочная система магазина цифровой техники
7. Информационно-справочная система обработки и доставки заказов пиццерии
8. Информационно-справочная система кинотеатра
9. Информационно-справочная система театра
10. Информационно-справочная система диллерского магазина автомобилей
11. Информационно-справочная система таксопарка
12. Информационно-справочная система кинопроката
13. Информационно-справочная система автосервиса
14. Информационно-справочная система автопроката
15. Информационно-справочная система перевозки грузов
16. Информационно-справочная система фитнес-клуба
17. Информационно-справочная система барбершопа

18. Информационно-справочная система массажного салона
19. Информационно-справочная система центра повышения квалификации
20. Информационно-справочная система для проведения научно-практических конференций
21. Информационно-справочная система для проведения конкурсов профессионального мастерства
22. Информационно-справочная система учебно-методических и правовых материалов компании
23. Информационно-справочная система учета инцидентов информационной безопасности в организации
24. Информационно-справочная система девелоперской компании
25. Информационно-справочная система по предоставлению услуг жилищно-коммунального хозяйства (ЖКХ)
26. Информационно-справочная система товарищества собственников жилья (ТСЖ)
27. Информационно-справочная система агентства недвижимости
28. Информационно-справочная система автотранспортного предприятия

VIII. Рекомендуемая литература

1. Козмина Ю., Харроп Р. Spring 5 для профессионалов. - Киев: Диалектика-Вильямс, 2019. - 1120 с.
2. Коузен К. Современный Java. Рецепты программирования . - М.: ДМК Пресс, 2018. - 274 с.
3. Мартин Роберт К. Чистый код. Создание анализ и рефакторинг. - СПб: Питер, 2019. - 464 с.
4. Прохоренок Н.А. JavaFX. - СПб: БХВ-Петербург, 2020. - 768 с.
5. Шилдт Г. Java. Полное руководство. - Киев: Диалектика, 2018. - 1488 с.

Учебное издание

Милованов Даниил Михайлович

ассистент Департамента анализа данных и машинного обучения факультета информационных технологий и анализа больших данных Финансового университета при Правительстве Российской Федерации

Хасаншин Ильшат Ядыкарович

кандидат технических наук, доцент Департамента анализа данных и машинного обучения факультета информационных технологий и анализа больших данных Финансового университета при Правительстве Российской Федерации

**МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ПО ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ ПО ДИСЦИПЛИНЕ «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ»
4 семестр**

Для студентов,
обучающихся по направлению
09.03.03 «Прикладная информатика»,
(программа подготовки бакалавра)

Компьютерный набор, верстка: Д.М. Милованов

Вычитка и корректура выполнены авторами

Электронное издание

© Финансовый университет при Правительстве Российской Федерации», 2021.

© Департамент анализа данных и машинного обучения, 2021.

© Милованов Д.И., Хасаншин И.Я., 2021.