

Министерство науки и высшего образования Российской Федерации

Калужский филиал
федерального государственного бюджетного образовательного
учреждения высшего образования
**«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»**
(КФ МГТУ им. Н.Э. Баумана)

С.А. Глебов, С.С. Гришунов

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Методические указания к выполнению
курсовой работы

Калуга – 2018

УДК 004.62
ББК 32.972.5
Г53

Методические указания составлены в соответствии с учебным планом КФ МГТУ им. Н.Э. Баумана по направлению подготовки 09.03.04 «Программная инженерия» кафедры «Программного обеспечения ЭВМ, информационных технологий».

Методические указания рассмотрены и одобрены:

- Кафедрой «Программного обеспечения ЭВМ, информационных технологий» (ИУ4-КФ) протокол № 2 от «26» сентября 2018 г.

Зав. кафедрой ИУ4-КФ  к.т.н., доцент Ю.Е. Гагарин

- Методической комиссией факультета ИУ-КФ протокол № 1 от «01» октября 2018 г.

Председатель методической комиссии факультета ИУ-КФ  к.т.н., доцент М.Ю. Адкин

- Методической комиссией КФ МГТУ им.Н.Э. Баумана протокол № 1 от «9» октября 2018 г.

Председатель методической комиссии КФ МГТУ им.Н.Э. Баумана  д.э.н., профессор О.Л. Перерва

Рецензент: к.т.н., доцент кафедры ИУ6-КФ  А.Б. Лачина

Авторы к.ф.-м.н., доцент кафедры ИУ4-КФ ассистент кафедры ИУ4-КФ  С.А. Глебов
С.С. Гришунов

Аннотация

Методические указания к выполнению курсовой работы по курсу «Объектно-ориентированное программирование» содержат сведения по методике написания, оформления и защиты курсовой работы по курсу «Объектно-ориентированное программирование».

Предназначены для студентов 3-го курса бакалавриата КФ МГТУ им. Н.Э. Баумана, обучающихся по направлению подготовки 09.03.04 «Программная инженерия».

© Калужский филиал МГТУ им. Н.Э. Баумана, 2018 г.
© С.А. Глебов, С.С. Гришунов, 2018 г.

ОГЛАВЛЕНИЕ

ОГЛАВЛЕНИЕ	3
ВВЕДЕНИЕ	4
ЦЕЛЬ И ЗАДАЧИ КУРСОВОЙ РАБОТЫ, ФОРМИРУЕМЫЕ НАВЫКИ	5
СТРУКТУРА КУРСОВОЙ РАБОТЫ	6
ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ И ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ	12
ОРГАНИЗАЦИЯ ЗАЩИТЫ КУРСОВОЙ РАБОТЫ	15
ТИПОВЫЕ ЗАДАНИЯ ДЛЯ КУРСОВОЙ РАБОТЫ.....	20
ПРИЛОЖЕНИЕ А.....	21
ПРИЛОЖЕНИЕ Б	22
ПРИЛОЖЕНИЕ В.....	24
ПРИЛОЖЕНИЕ Г	27
ПРИЛОЖЕНИЕ Д.....	31
ЛИТЕРАТУРА	32

ВВЕДЕНИЕ

Курсовая работа является важным элементом учебного процесса подготовки высококвалифицированного специалиста. Она должна быть самостоятельным научно-практическим исследованием студента, имеющим целью закрепить и систематизировать знания, полученные в период обучения в целом и по избранной теме в частности; развить умения и научить применять полученные знания для решения конкретных научных и практических проблем, формулировать и аргументировать собственную позицию в их решении. Курсовая работа представляет собой разработку структуры классов приложения, способствующих углубленному изучению основных, наиболее трудных и важных разделов дисциплины "Объектно-ориентированное программирование".

Данные указания предоставляют студенту возможность правильно и квалифицированно выполнить курсовую работу, соблюдая при этом все стандарты и требования по её оформлению. Хорошо подготовленная и успешно защищенная курсовая работа дает студенту уверенность в своей подготовленности к выполнению в будущем выпускной квалификационной работы, качество которой повышается, если она является завершением самостоятельно и качественно выполненных студентом курсовых проектов и работ.

Предназначены для студентов 3-го курса бакалавриата КФ МГТУ им Н.Э. Баумана, обучающихся по направлению подготовки 09.03.04 «Программная инженерия».

ЦЕЛЬ И ЗАДАЧИ КУРСОВОЙ РАБОТЫ, ФОРМИРУЕМЫЕ НАВЫКИ

Целью курсовой работы является формирование практических навыков по разработке и реализации структуры классов

Задачи проектирования:

1. Овладение первичными навыками ведения научно-исследовательской, проектной и производственно-технологической деятельности, развитие творческих способностей индивидуально для каждого студента.

2. Усвоение методов грамотного ведения, оформления и редактирования технической документации.

При выполнении курсовой работы у студентов формируются следующие навыки:

- самостоятельной научно-исследовательской деятельности;
- грамотного оформления полученных результатов;
- представления результатов своей работы в виде доклада;
- защиты полученных результатов в дискуссии.

СТРУКТУРА КУРСОВОЙ РАБОТЫ

Основные требования к структуре

Курсовая работа содержит две части — расчетно-пояснительную записку и графическую часть. Данные компоненты оформляются каждым студентом индивидуально и содержат описание лично выполненной работы. Расчетно-пояснительная записка курсовой работы по разработке программного приложения состоит из следующих частей:

1. Техническое задание.
2. Методы и инструменты программной инженерии.
3. Проектирование компонентов программного продукта.
4. Тестирование и интеграция компонентов программного продукта.

5. Заключение.

6. Список использованных источников.

7. Приложения

Содержание разделов, состав подразделов и пунктов расчетно-пояснительной записки определяются темой и характером курсовой работы.

Графическая часть работы, оформленная на листах формата А1, наглядно иллюстрирует выполненную работу. Оформление основных элементов схем алгоритмов показано в [Приложении А](#). Расчетно-пояснительная записка обязательно должна содержать конструкторскую и/или технологическую документацию, сопровождающую спроектированное информационно-программное изделие и иллюстративный материал, необходимый студенту при защите курсовой работы. В графической части должны быть приведены чертежи, схемы, плакаты, отражающие:

- архитектуру разработанного приложения;
- UML-диаграмма классов;
- процесс обработки информации (спецификации, алгоритмы и т.п.);
- структуру и взаимодействие модулей приложения;
- схемы баз данных;
- описание интерфейса взаимодействия;
- демонстрационные плакаты, которые помогут студенту при защите лучше раскрыть замысел разработанного приложения.

Выполнение курсовой работы осуществляется студентом на основе задания на курсовую работу. Задание на курсовую работу определяет общие требования на состав и содержание курсовой работы как на научно-техническую продукцию, каковой и является результат выполнения курсовой работы.

Форма задания на курсовую работу утверждена УМУ МГТУ и содержит следующие разделы:

- Наименование дисциплины.
- ФИО студента и руководителя.
- График выполнения курсовой работы.
- Тема работы.
- Техническое задание.
- Перечень графического материала.

Титульный лист и бланк задания на курсовую работу приведены в [Приложении Б](#).

Задание на курсовую работу подписывается руководителем работы, студентом и утверждается заведующим кафедрой. Техническое задание на программное приложение, разрабатываемое в рамках курсовой работы, составляется и оформляется в соответствии с ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы».

Учитывая учебный характер курсовой работы, можно рекомендовать следующий состав технического задания на разработку автоматизированной системы обработки информации:

1. Наименование.
2. Основание для разработки.
3. Исполнитель.
4. Цель разработки.
5. Содержание работы:
 - 5.1. Задачи, подлежащие решению.
 - 5.2. Требования к архитектуре программного приложения.
 - 5.3. Требования к составу программных компонентов.
 - 5.4. Требования к прикладным программам.
 - 5.5. Требования к входным/выходным данным.
 - 5.6. Требования к временным характеристикам.
 - 5.7. Требования к составу технических средств.
6. Этапы разработки.

7. Техническая документация, предъявляемая по окончании работы.

8. Дополнительные условия.

Рассмотрим более подробно примерное содержание остальных разделов расчетно-пояснительной записки курсовой работы.

Методы и инструменты программной инженерии

Материалы этой части могут быть расположены в расчетно-пояснительной записке в следующих подразделах:

2.1 Постановка задачи проектирования.

2.2 Описание предметной области.

2.3 Анализа налогов и прототипов.

2.4 Перечень задач, подлежащих решению в процессе разработки.

2.5 Обоснование выбора инструментов и платформы для разработки.

Подробная характеристика основных подразделов приведена ниже.

Описание предметной области. Этот подраздел должен содержать:

- естественно-языковую модель предметной области (описание предметной области на русском языке);
- выделенные в процессе анализа предметной области сущности и отношения между ними;
- существующие технологии обработки информации, включая способы её передачи и обеспечения безопасности;
- перечень функций, подлежащих автоматизации;
- наглядно-графическую модель предметной области.

Анализ аналогов и прототипов. В подразделе приводятся описания аналогов и прототипов создаваемого приложения, выявленных при работе с литературными источниками, исследований аналогичных разработок, приводятся сравнительные результаты анализа аналогов и тенденций развития.

Перечень задач, подлежащих решению в процессе разработки. В подразделе следует перечислить все задачи, стоящие перед проектировщиком, которые он должен решить в процессе разработки приложения и его отладки, включая проведение необходимых исследований при создании программного продукта.

Проектирование компонентов программного продукта

Материалы этого подраздела содержат результаты работ, выполняемых на этапах эскизного, технического и рабочего проектирования, что соответствует проектированию программно-информационного комплекса.

Результаты проектирования могут быть расположены в расчетно-пояснительной записке в следующих подразделах:

- 3.1. Разработка структуры приложения.
- 3.2. Разработка структуры классов приложения.
- 3.3. Разработка базы данных.
- 3.4. Разработка алгоритмов обработки информации.
- 3.5. Разработка архитектуры приложения.
- 3.6. Разработка интерфейса взаимодействия пользователя с системой.

Материалы проектной части отражаются в графической части курсовой работы, оформленные в виде чертежей и схем. Все принимаемые технические решения должны быть грамотно обоснованы.

Количество и названия подразделов определяются студентом самостоятельно, с целью максимально эффективного освещения процесса проектирования приложения.

Приведем перечни вопросов, которые должны быть отражены в каждом из подразделов данной части курсовой работы.

Разработка структуры приложения. Результатом внешнего проектирования является содержательный облик приложения, который оформлен в виде естественно-языкового описания. На основе этой модели и разрабатывается структура приложения (структурная схема системы). Она позволяет:

- выявить источники информации и её потребителей;
- определить состав компонентов системы, СУБД, среду и платформу для разработки;
- наметить последовательность разработки системы и её компонентов.

Разработка структуры классов приложения. Результатом разработки структуры классов является UML-диаграмма классов, отражающая основной набор классов, их свойств и методов, структуру наследования классов, реализации интерфейсов. Основные правила оформления UML-диаграммы классов приведены в [Приложении В](#).

Разработка базы данных. Результатом проектирования базы данных является ER-диаграмма, отражающая содержание таблиц и связей между ними. Также необходимо привести описание используемых для обработки данных хранимых процедур и триггеров, созданных просмотров, генераторов и созданных типов данных.

Разработка алгоритмов обработки информации. Для качественной обработки информации в программном приложении необходимо:

- разработать алгоритмы решения функциональных задач;
- провести тестирование алгоритмов на точность, непротиворечивость, конечность и другие критерии качества.

В процессе проектирования алгоритмов определяются требуемые библиотеки и функции, время и точность решения предлагаемых функциональных задач.

Разработанные алгоритмы должны быть представлены в графической части работы в виде алгоритмических схем.

Разработка архитектуры приложения. В подразделе помещаются материалы по выбору состава и номенклатуры комплекса технических средств (КТС) и общесистемных пакетов. Для обоснованного выбора предпочтительного варианта технического решения необходимо:

- определить (разработать) варианты воплощения программного продукта;
- оценить предполагаемое качество функционирования системы или её компонентов по результатам моделирования на моделях различного уровня;
- выбрать наилучший вариант реализации.

Разработка интерфейса взаимодействия пользователя с системой. При разработке интерфейса взаимодействия следует обратить особое внимание на вид деятельности пользователя, для автоматизации которой применяется разрабатываемое приложение. Разработка-интерфейса взаимодействия включает в себя создание:

- технологии взаимодействия пользователя с системой (обоснованного с различных точек зрения интерфейса взаимодействия);
- информационных моделей отображения информации на экране монитора (меню, текстовая и графическая информация о результатах решения функциональных задач, результаты выполнения процедур и т. п.);

— графа взаимодействия пользователя с системой.

При реализации спроектированных информационных моделей и интерфейса взаимодействия необходимо обратить особое внимание на эргономические требования к форме отображения информации на экране монитора, учитывая не только психофизиологические и антропометрические характеристики человека, но и его мыслительную, эмоциональную и эстетическую сферу деятельности. Также необходимо разработать требования к профессиональному уровню пользователя программного продукта.

Тестирование и интеграция компонентов программного продукта

В этом разделе курсовой работы приводятся материалы, которые должны доказать, что предложенные проектные решения могут быть реализованы. Доказательства возможности реализации программного приложения могут быть приведены в следующих подразделах:

4.1. Тестирование и отладка макета рабочей программы.

4.2. Разработка руководства пользователя и руководства программиста (администратора).

4.3. Экспериментальные данные тестирования процедур и функциональных задач.

Тестирование и отладка макета рабочей программы. Данная стадия рабочего проекта завершает проектирование программного продукта. Она заключается в отладке макета программы или программы-оригинала и формулировке обоснованных требований к аппаратной и программной платформе, необходимых для успешной эксплуатации разработанной системы.

Разработка руководства пользователя и руководства администратора(программиста). Так как в рамках курсовой работы не представляется возможным выпускать полный комплект рабочей документации на разработанный программный продукт, то объем выпускаемой документации ограничен руководствами пользователя и программиста, эксплуатирующих систему.

Экспериментальные данные тестирования процедур и функциональных задач. Материалы тестирования оформляются после описания тестовых примеров в виде таблиц, графиков, снимков экрана с возможным вынесением их в графическую часть курсовой работы.

Необходимо также отметить перспективы или факты внедрения разработанного программного приложения.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ И ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ

Студент выбирает тему, предложенную руководителем, или определяет тему самостоятельно, но обосновав при этом целесообразность её разработки. Совпадение тем курсовых работ у студентов одной учебной группы недопустимо.

При выборе темы следует руководствоваться:

- личным интересом к ней, осознанием её актуальности, важности;
- возможностью повышения компетентности и профессионализма в ходе подготовки работы;
- возможностью применения полученных знаний на практике в настоящей и будущей деятельности.

Формулировка темы должна конкретно отображать суть разрабатываемого приложения, она должна быть по возможности краткой и соответствовать содержанию работы.

Темы курсовых работ определяются не позднее двух недель с начала учебного семестра. День получения темы и задания фиксируется в специальном журнале. Вся работа по выполнению курсовой разбивается на 4 этапа, каждый из которых составляет 25% общего объема работы. Этапы выполнения курсовой работы перечислены в табл. 1.

Таблица 1 - Этапы выполнения курсовой работы

№ Этапа	Период этапа	Результат деятельности
1	1-4 недели	Выбор и согласование темы с руководителем; проведение исследования по выбранной теме, обзор существующих программных продуктов; разработка технического задания.
2	5-7 недели	Разработка и реализация алгоритмов функционирования приложения, структуры, систем передачи информации, технологий обработки информации и интерфейса взаимодействия пользователя с системой. Выполнение и оформление проектной части работы
3	9-10 недели	Тестирование и отладка программного приложения. Разработка руководства пользователя и программиста. Написание производственно-технологической части расчетно-пояснительной записки
4	11-14 недели	Завершающее оформление документации согласно требованиям ГОСТ и данного методического пособия. Подготовка доклада. Защита курсовой работы

Возможный вариант оформления «Расчетно-пояснительной записки».

Расчетно-пояснительная записка — это документ, который в четкой и краткой форме раскрывает творческий замысел и его реализацию. В записке отражаются этапы работы и результаты, полученные при её выполнении. Материалы в расчетно-пояснительной записке к курсовой работе должны быть расположены в следующем порядке:

Титульный лист с подписями.

Утвержденное задание на курсовую работу.

Содержание (с постраничной разметкой).

Основная часть, разделенная на разделы:

- методы и инструменты программной инженерии.
- проектирование компонентов программного продукта.
- тестирование и интеграция компонентов программного продукта.

Заключение.

Список использованных источников.

Приложения.

Объем расчетно-пояснительной записки — 30 – 40 страниц. В ней должны содержаться рисунки, схемы, графики и т. п. в таком объеме, чтобы записка могла быть прочитана независимо от графической части работы. Расчетно-пояснительная записка выполняется с учетом требований, предусмотренных ГОСТ 7.32-2001 и 2.105-95. Оформление рисунков, таблиц и формул показано в [Приложении Г](#).

Текст РПЗ печатается на одной стороне листа белой бумаги формата А4 через полтора интервала. Цвет шрифта – черный. Размер шрифта (кегель) – 12-14. Обычная практика – кегль 14. ГОСТ не определяет тип шрифта, но обычно – TimesNewRoman.

В случае наличия программного кода, рекомендуется использовать размер шрифта (кегель) – 10, тип шрифта – CourierNew.

Размеры полей: правое – 10 мм, верхнее и нижнее – 20 мм, левое – 30 мм.

Страницы работы нумеруются арабскими цифрами (нумерация сквозная по всему тексту). Номер страницы ставится в центре нижней части листа без точки. Титульный лист и бланк задания включаются в общую нумерацию, номер на них не ставится.

Размер абзацного отступа составляет 15 мм.

Расстояние между заголовком и текстом равно одной пустой строке.

По ГОСТ 7.32-2001 заголовок СОДЕРЖАНИЕ пишется заглавными буквами посередине строки.

Содержание включает введение, наименование всех глав, параграфов, пунктов, заключение, список использованных источников и наименование приложений с указанием номеров страниц, с которых начинаются эти элементы работы.

По ГОСТ 2.105-95 наименования, включенные в содержание, записывают строчными буквами, начиная с прописной буквы.

Ориентировочное распределение объема работы по разделам в страницах:

- техническое задание — 3-5;
- методы и инструменты программной инженерии — не менее 8;
- проектирование компонентов программного продукта — не менее 8;
- тестирование и интеграция компонентов программного продукта — не менее 6;

При разработке графической части курсовой работы необходимо руководствоваться требованиями стандартов ЕСКД и ЕСПД с отступлениями, установленными в них для учебных заведений (Пример оформления штампа приведен в [Приложении Д](#)).

Для более наглядного показа всех видов выполненной студентом работы в графическую часть включаются плакаты, диаграммы, графики, таблицы с результатами экспериментов и т. п.

Графическая часть курсовой работы должна быть выполнена с применением специализированного программного обеспечения (для схем алгоритмов желательно использовать MSVisio).

В приложение к расчетно-пояснительной записке выносятся таблицы с результатами экспериментов, тексты программ и другие программные документы, разработанные в ходе выполнения курсовой работы. Также в приложение могут быть включены различные материалы, по мнению разработчика, необходимые для правильной и полной оценки курсовой работы. Листинг программных модулей печатается с использованием шрифта *CourierNew* размером 8-10 пт. По согласованию с руководителем курсовой работы, также допускается запись программного кода на цифровой носитель информации.

Для проверки оформления расчетно-пояснительной записки можно воспользоваться приложением [TestVkr](#).

ОРГАНИЗАЦИЯ ЗАЩИТЫ КУРСОВОЙ РАБОТЫ

Защита представляет собой завершающий этап выполнения курсовой работы. Цель её — выявить глубину знаний студента по исследуемой проблеме и самостоятельность выполнения работы. Студент должен хорошо ориентироваться в представленной работе; уметь раскрыть функциональное назначение программно-информационного комплекса на основе разработанной структуры и алгоритмов; знать назначение основных функций приложения, как системных, так и разработанных, их входные и выходные параметры; ответить на вопросы как теоретического, так и практического характера, которые могут возникнуть при обсуждении данной работы. Для защиты курсовых работ утверждается график работы комиссии. Время и место приема курсовых работ доводится до сведения студентов не позднее чем за одну неделю до защиты. Для приема защиты курсовой работы кафедра формирует комиссию, в состав которой должно входить не менее трех преподавателей кафедры.

На каждом заседании комиссии целесообразно заслушивать не менее 3 и не более 8 работ. Защита курсовой работы происходит в следующей последовательности:

- доклад студента (не более 5-7 мин);
- ответы студента на вопросы комиссии.

Доклад начинается с представления докладчика и названия курсовой работы. В докладе студенту следует изложить важнейшие этапы, особенности и результаты работы, не вдаваясь скрупулезно в тонкости конкретных технических решений, четко сформулировать выводы по выполненной работе.

В докладе необходимо отметить:

- актуальность выбранной темы;
- цели и задачи работы;
- степень решения поставленных задач;
- полученные результаты;
- выводы и предложения по существу выполненной работы;
- возможность практического использования полученных результатов.

Заканчивается доклад фразой «Доклад завершён, спасибо за внимание».

На защиту не допускаются работы, которые не были продемонстрированы руководителю, а также работы имеющие высокий (более 40%) процент заимствований в материалах работы.

Вопросы, задаваемые студенту по докладу, могут касаться деталей выполненной работы либо общих теоретических положений, связанных или не связанных с темой работы, в пределах изученной учебной программы по курсу «Базы данных».

Типовые вопросы и задания для защиты работы:

Оценка знаний

- Перечислите и раскройте задачи курсовой работы.
- Опишите назначение и область применения разработанного программного продукта.

Оценка умений

- Назовите использованные в работе паттерны объектно-ориентированного проектирования.
- Раскройте методику реализации графического интерфейса программного обеспечения на примере выполненной работы.
- Раскройте методику выбора языка и средств разработки на примере выполненной работы.

Оценка навыков

- Оцените отказоустойчивость разработанного программного продукта.
- Раскройте пути для обеспечения требований к надежности и безопасности программного обеспечения на примере выполненной работы.

Решение об оценке курсовой работы принимается по результатам анализа представленной работы, доклада студента, правильного оформления документации в соответствии с ГОСТ и ответов студента на вопросы. Оценка по итогам защиты курсовой работы проставляется в ведомость и зачетную книжку студента научным руководителем.

Оценка за курсовую работу складывается из следующих оценок:

- оценка качества работы руководителем;
- оценка своевременности представления работы к защите;
- оценка качества защиты.

Максимальная семестровая оценка за выполненную курсовую работу составляет 100 баллов, минимальная оценка – 60 баллов.

Оценивание *качества* выполнения курсовой работы баллами производится руководителем по следующим критериям:

Таблица 2

Баллы	Критерии
43-50	Задание выполнено в полном объеме. В ходе выполнения курсовой работы студент продемонстрировал самостоятельность и аргументированность принятия решений. Требования к объему курсовой работы выполнены полностью, либо превосходят их. Выполнены установленные требования по оформлению расчетно-пояснительной записки и графической части работы.
38-42	Задание выполнено в полном объеме. В ходе выполнения курсовой работы студент не всегда демонстрировал самостоятельность и аргументированность принятия решений. Требования к объему курсовой работы выполнены полностью. В основном выполнены установленные требования по оформлению расчетно-пояснительной записки и графической части работы.
30-37	Задание на курсовую работу выполнено в полном объеме. В ходе выполнения курсовой работы студент продемонстрировал самостоятельность и аргументированность большинства принятия решений. Требования к объему курсовой работы выполнены полностью. Имеются некритические отступления от установленных требований по оформлению расчетно-пояснительной записки и графической части работы.
0	Задание на курсовую работу выполнено в неполном объеме. В ходе выполнения курсовой работы студент не смог продемонстрировать самостоятельность и/или аргументированность принятия решений. Имеются критические отступления от установленных требований по оформлению расчетно-пояснительной записки и графической части работы.

Оценивание своевременности представления работы к защите производится комиссией по следующим критериям:

Таблица 3

Баллы	Критерии
7	Курсовая работа представлена на защиту в установленное время.
5	Курсовая работа представлена на защиту с отставанием от графика, но до начала экзаменационной сессии
0	Курсовая работа представлена на защиту в период экзаменационной сессии

Оценивание качества защиты курсовой работы комиссией производится по следующим критериям:

Таблица 4

Баллы	Критерии
40-43	Доклад длится 5-7 мин. В ходе защиты автор уверенно и аргументировано отвечает на задаваемые комиссией вопросы, показывает глубину и полную проработку рассматриваемой предметной области.
35-39	Длительность доклада близка к установленному регламенту. В ходе защиты студент достаточно полно и обоснованно отвечает на вопросы и замечания членов комиссии. Процесс защиты демонстрирует достаточно проработку тематики курсовой работы. Вместе с тем, в ходе доклада и дискуссии в речи и ответах на вопросы имеется ряд недостатков, не имеющих принципиального характера, но свидетельствующих о недоскональной проработке материала.
30-34	Длительность доклада отличается от установленного регламента. В ходе защиты студент демонстрирует слабые знания предметной области. Речь содержит значительные фактические ошибки. Ответы на вопросы членов комиссии свидетельствуют о пробелах в понимании как междисциплинарных связей, так и показывают фрагментарный характер знаний студента по дисциплине.
0	Длительность доклада значительно отличается от установленного регламента. В процессе защиты неуверенно и логически непоследовательно излагается материал, отсутствует план доклада, студент не понимает сущности задаваемых вопросов, в ответах присутствуют грубые ошибки или студент затрудняется с ответом.

Перевод набранной суммы баллов в оценку по дисциплине производится по таблице 5.

Таблица 5

Диапазон баллов	Шкала оценок
90-100	Отлично
75-89	Хорошо
60-74	Удовлетворительно
0-59	Неудовлетворительно

В случае неудовлетворительной оценки комиссия может предложить проведение дополнительной проработки отдельных вопросов и установить срок представления работы к защите после доработки.

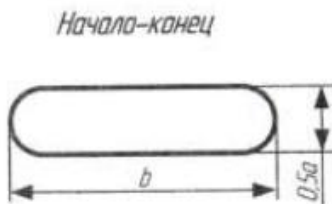
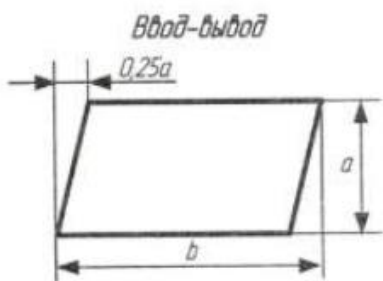
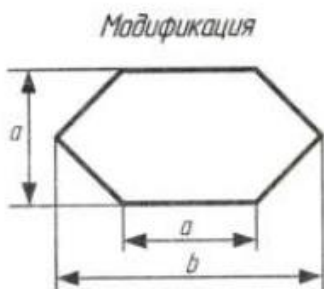
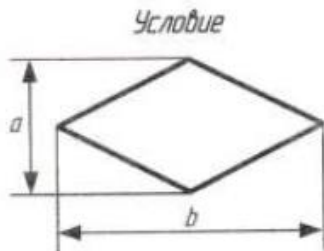
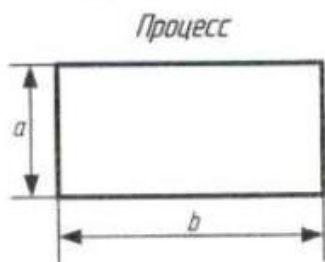
По результатам защиты лучшие курсовые работы могут быть рекомендованы кафедрой для опубликования в сборниках студенческих работ, издаваемых в вузе.

ТИПОВЫЕ ЗАДАНИЯ ДЛЯ КУРСОВОЙ РАБОТЫ

- Разработать игровое приложение «Змейка».
- Разработать систему для составления расписания в вузе.
- Разработать информационную систему записи пациентов на прием к врачу.
- Разработать информационную систему учета транспорта.
- Разработать информационную систему бронирования билетов.
- Разработать информационную систему службы доставки товаров.
- Разработать персональную медиа-библиотеку.
- Разработать новостной сайт.
- Разработать редактор изображений.
- Разработать редактор кода.
- Разработать приложение для сжатия и шифрования файлов.

ПРИЛОЖЕНИЕ А

Некоторые условные обозначения схем алгоритмов согласно ГОСТ 19.003-80. Соотношение сторон элементов (a , b) выбирается либо 2 к 1, либо 1,5 к 1.



ПРИЛОЖЕНИЕ Б



Министерство образования и науки Российской Федерации
Калужский филиал федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____

КАФЕДРА _____

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе на тему:

Студент _____ (_____) (ф.и.о.)

Руководитель курсовой работы _____ (_____) (ф.и.о.)

Калуга, 20 ____

Каждый филиал федерального государственного бюджетного образовательного учреждения
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ
Заведующий кафедрой _____
(_____)
«___» _____ 20__ г.

ЗАДАНИЕ на выполнение курсовой работы

по дисциплине _____

Студент _____
(фамилия, инициалы, индекс группы)

Руководитель _____
(фамилия, инициалы)

График выполнения работы: 25% к _____ нед., 50% к _____ нед., 75% к _____ нед., 100% к _____ нед.

1. Тема курсовой работы

2. Техническое задание

3. Оформление курсовой работы

3.1. Расчетно-поисковая записка на _____ листок формата А4.

3.2. Перечень графического материала КП (плакаты, схемы, чертежи и т.п.) _____

Дата выдачи задания «_____» _____ 20__ г.

Руководитель курсовой работы _____ / _____ /

Задание получил _____ / _____ / «_____» _____ 20__ г.

Примечание:

1. Задание оформляется в двух экземплярах; один выдается студенту, второй хранится на кафедре.

ПРИЛОЖЕНИЕ В

Диаграмма класса описывает типы объектов системы и различного рода статические отношения, которые существуют между ними. Также отображаются свойства классов, операции и ограничения, накладываемые на связи между объектами.

Графически классы на диаграмме отображаются прямоугольниками, разделенными на 3 части: имя класса (жирный шрифт), его атрибуты и операции.

Атрибут описывает свойство в виде строки текста. Полная форма атрибута:

<видимость> <имя>: <тип> <кратность> = <значение по умолчанию> {<свойства>}

Обязательным является только имя.

Видимость может принимать одно из следующих значений:

- + – открытый (public);
- закрытый (private);
- ~ – пакетный (package);
- # – защищенный (protected).

Кратность свойства обозначает количество объектов, которые могут заполнять данное свойство. В большинстве случаев кратности определяются верхними и нижними границами, например 2..5, если нижняя и верхняя границы совпадают, то указывается одно число; символ «*» обозначает количество без ограничений.

Свойства могут принимать такие значения как *frozen*, *read-only*, *abstract*, *ordered* и др.

Операция представляет собой действие, реализуемое классом. Операции, просто манипулирующие свойствами, не указываются. Полный синтаксис операций выглядит следующим образом:

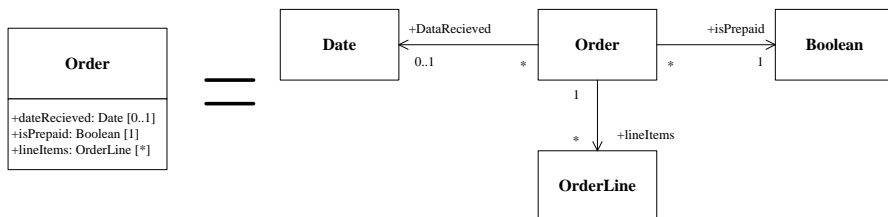
<видимость> <имя> (<список параметров>): <возвращаемый тип> {<строка свойств>}

Параметры в списке параметров обозначаются аналогично атрибутам.

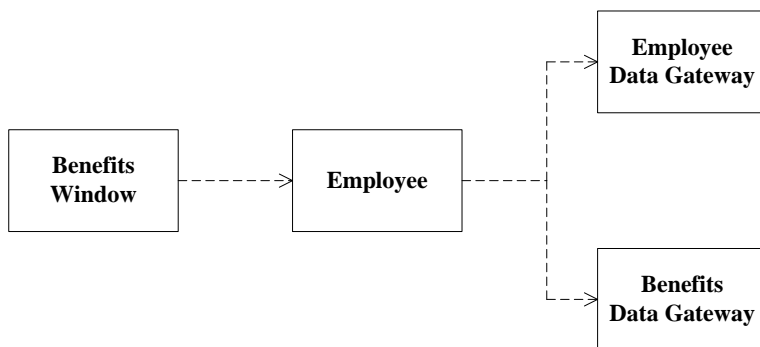
Статические атрибуты и операции на диаграммах выполняются с подчеркиванием.

Абстрактные операции выполняются курсивом.

Значительную часть атрибута можно изобразить в ассоциации. Ассоциация изображается непрерывной линией со стрелкой на конце, направленная от исходного класса к целевому. Также возможны двусторонние ассоциации.

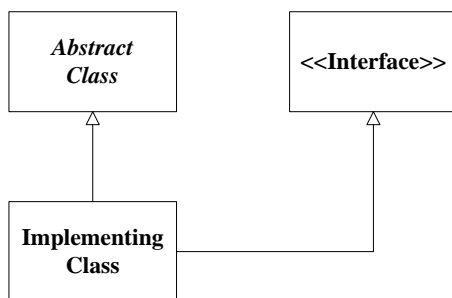


Считается, что между двумя элементами существует **зависимость**, если изменение в определении одного из них (сервера) могут вызвать изменения в другом элементе (клиенте). Зависимость на диаграмме изображается пунктирной линией со стрелкой на конце от клиента к серверу.



Обобщение (наследование) на диаграммах изображается непрерывной линией с пустой стрелкой на конце.

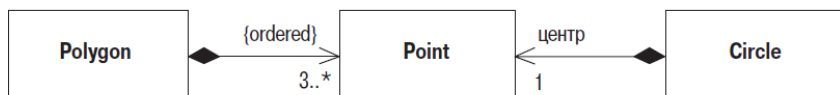
Интерфейсы на диаграммах заключаются во французские кавычки («елочки»). Реализация интерфейсов на диаграммах обозначается пунктирной линией с пустой стрелкой на конце.



Агрегация – это отношения типа «часть целого». Отображается с использованием незакрашенного ромба напротив класса-целого.



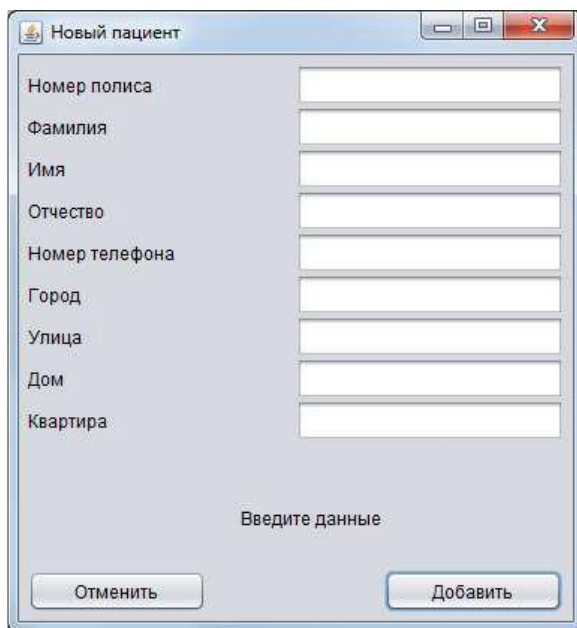
Композиция схожа с агрегацией. Отличие в том, что хотя класс может быть частью нескольких других классов, любой экземпляр может принадлежать только одному владельцу. Изображается с использованием закрашенного ромба напротив класса-владельца.



ПРИЛОЖЕНИЕ Г

Оформление рисунков

По ГОСТ 7.32-2001 на все рисунки в тексте должны быть даны ссылки. Рисунки должны располагаться непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. Выравнивание рисунка осуществляется по центру строки. Рисунки нумеруются арабскими цифрами, при этом нумерация сквозная, но допускается нумеровать и в пределах раздела (главы). В последнем случае номер рисунка состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой (например: Рисунок 1.1). Подпись к рисунку располагается под ним посередине строки. Слово «Рисунок» пишется полностью. По ГОСТу можно ограничиться только номером (т.е. оставить, например, подпись: Рисунок 2), но практически всегда требуется название. В этом случае подпись должна выглядеть так:



The image shows a screenshot of a software window titled "Новый пациент" (New Patient). The window has a standard Windows-style title bar with minimize, maximize, and close buttons. Inside the window, there is a list of labels on the left and corresponding text input fields on the right. The labels are: "Номер полиса" (Policy Number), "Фамилия" (Surname), "Имя" (Name), "Отчество" (Patronymic), "Номер телефона" (Phone Number), "Город" (City), "Улица" (Street), "Дом" (House), and "Квартира" (Apartment). Below these fields, there is a text label "Введите данные" (Enter data). At the bottom of the window, there are two buttons: "Отменить" (Cancel) on the left and "Добавить" (Add) on the right.

Рисунок 8 – Меню добавления пациента

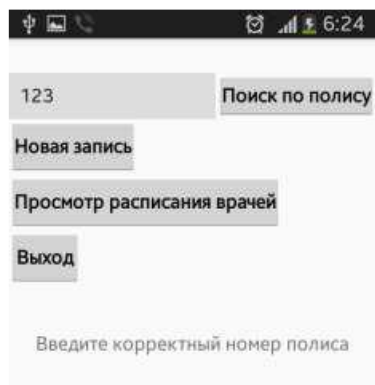


Рисунок 3.2 – Интерфейс мобильного приложения

Точка в конце названия рисунка не ставится. Если в работе есть приложения, то рисунки каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением впереди обозначения приложения (например: Рисунок А.3).

Оформление таблиц

По ГОСТ 7.32-2001 на все таблицы в тексте должны быть ссылки. Таблица должна располагаться непосредственно после текста, в котором она упоминается впервые, или на следующей странице. Допускается выравнивание по левому краю или по середине строки. Все таблицы нумеруются (нумерация сквозная, либо в пределах раздела – в последнем случае номер таблицы состоит из номера раздела и порядкового номера внутри раздела, разделенных точкой (например: Таблица 1.2)). Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением впереди обозначения приложения (например: Таблица В.2). Слово «Таблица» пишется полностью. Наличие у таблицы собственного названия по ГОСТу не обязательно. Если имеется название таблицы, то его следует помещать надтаблицей слева, без абзачного отступа в одну строку с ее номером через тире (например: Таблица 3 – Доходы фирмы). Точка в конце названия не ставится.

Таблица 1.1 – Стадии и этапы разработки ТЗ

Стадии разработки	Этапы работ	Содержание работ	Сроки
Техническое задание	Обоснование необходимости разработки программы	Постановка задачи. Сбор исходных материалов. Выбор и обоснование критериев эффективности и качества разрабатываемой программы.	08.02.2016 - 15.02.2016
	Разработка и утверждение технического задания	Определение требований к программе. Разработка технико-экономического обоснования разработки программы. Определение стадий, этапов и сроков разработки программы и документации на нее. Выбор языков программирования. Согласование и утверждение технического задания.	16.02.2016 - 25.02.2016

При переносе таблицы на следующую страницу название помещают только над первой частью, при этом нижнюю горизонтальную черту, ограничивающую первую часть таблицы, не проводят. Над другими частями также слева пишут слово «Продолжение» и указывают номер таблицы (например: Продолжение таблицы 1). Заголовки столбцов и строк таблицы следует писать с прописной буквы в единственном числе, а подзаголовки столбцов – со строчной буквы, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют самостоятельное значение. В конце заголовков и подзаголовков столбцов и строк точки не ставят.

Разделять заголовки и подзаголовки боковых столбцов диагональными линиями не допускается.

Заголовки столбцов, как правило, записывают параллельно строкам таблицы, но при необходимости допускается их перпендикулярное расположение.

Горизонтальные и вертикальные линии, разграничивающие строки таблицы, допускается не проводить, если их отсутствие не затрудняет пользование таблицей.

ПРИЛОЖЕНИЕ Д

На графических листах в основной надписи необходимо использовать обозначения по следующему шаблону:

К.Р. №ЗКГЗ - 01 КД №Д - 2,

где №ЗК — номер зачетной книжки; ГЗ — год защиты; 01 — номер редакции документа; КД — код вида документа по ГОСТ 19.101, например: 81 — схема алгоритма, 90 — схема структурная, 91 — структура БД, 92 — графики, 98 — демонстрационные чертежи; №Д — номер документа данного вида; 2 — номер части документа (если из одной части, то не указывается). Таким образом, получаем обозначение типа

К.Р. 12Ф534 16 - 01 90 01.

Пример оформления основной надписи:

				К.Р. 05Ф007 06 — 01 98 01		
Имя	Лист	№ докум	Подп	Дата	Демонстрационный	
Разработ		Иванов А.А.			чертеж	
Проб		Петров А.А.				
Г. контрол						
Н. контрол						
Зам		Петров А.А.				
					Лист	Масса
					Лист 1	Листов 3
					КФ МГТУ	
					ИТД 72	

фамилия студента фамилия руководителя Название листа Номер листа Индекс группы Всего листов

ЛИТЕРАТУРА

1. Романенко, В.В. Объектно-ориентированное программирование: учебное пособие / В.В. Романенко; Министерство образования и науки Российской Федерации, Томский Государственный Университет Систем Управления и Радиоэлектроники (ТУСУР). - Томск: Томский государственный университет систем управления и радиоэлектроники, 2014. - 475 с. : ил. - Библиогр.: с. 442. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=480517> (24.08.2018).

2. Николаев, Е.И. Объектно-ориентированное программирование: учебное пособие / Е.И. Николаев ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Северо-Кавказский федеральный университет». - Ставрополь: СКФУ, 2015. - 225 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=458133> (24.08.2018).

3. Суханов, М.В. Основы Microsoft .NET Framework и языка программирования C#: учебное пособие / М.В. Суханов, И.В. Бачурин, И.С. Майоров ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего профессионального образования Северный (Арктический) федеральный университет им. М.В. Ломоносова. - Архангельск: ИД САФУ, 2014. - 97 с.: схем., табл., ил. - Библиогр. в кн. - ISBN 978-5-261-00934-4 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=312313> (24.08.2018).