

Методические рекомендации
по выполнению курсового проекта
по МДК 02.01 "Технология разработки программного обеспечения"
для специальности 09.02.07 «Информационные системы и
программирование»

Отделение	Индустриальных и цифровых технологий
ПЦК	Информационных технологий и программирования
Курс	4
Семестр	8

Методические рекомендации по выполнению курсового проекта по МДК 02.01 «Технология разработки программного обеспечения» составлены в соответствии с ФГОС СПО по специальности 09.02.07 «Информационные системы и программирование», утвержденным Минобрнауки России от 9 декабря 2016 г. № 1547.

ОДОБРЕНА

На заседании ПЦК Информационных
технологий и программирования
«1» декабря 2022 г., Протокол № 3

Председатель ПЦК ИТиП _____ А.А. Костина

«__» _____ 2022 г.

РАЗРАБОТАЛ

Преподаватель _____ А.А. Костина

СОГЛАСОВАНО

Зав. отделением ИЦТ _____ О.В. Бойко

СОДЕРЖАНИЕ

1. Общие положения.....	4
2 Основные этапы выполнения курсового проекта	5
3 Требования к содержанию курсового проекта	6
Содержание.....	6
Введение.....	7
Анализ предметной области	9
Разработка технического задания (или требований к программному обеспечению).....	10
Выбор модели жизненного цикла ПО	11
Проектирование программного обеспечения	12
Реализация программного обеспечения с помощью языка программирования (или программного средства).....	14
Тестирование и отладка программного обеспечения	17
Экономическая оценка программного обеспечения	18
Заключение	19
Библиография (список использованной литературы).....	19
4 Оформление курсового проекта	21
4.1 Общие требования	21
4.2 Ссылки.....	22
4.3. Основные элементы библиографического описания	22
4.4. Правила сокращения слов и словосочетаний	23
5. Аттестация курсового проекта	25
ПРИЛОЖЕНИЕ 1	26
ПРИЛОЖЕНИЕ 2	27
ПРИЛОЖЕНИЕ 3	28
ПРИЛОЖЕНИЕ 4.....	29

1. Общие положения

Курсовой проект по МДК 02.01 «Технология разработки программного обеспечения» предусмотрен учебным планом специальности 09.02.07 «Информационные системы и программирование» и является итоговым контролем знаний студентов.

Выполнение курсового проекта является неотъемлемой и необходимой частью изучения МДК 02.01 «Технология разработки программного обеспечения».

Цель курсового проекта – научить студентов самостоятельно применять знания по МДК 02.01 «Технология разработки программного обеспечения» для решения конкретных практических задач в области технологий создания программных продуктов.

Выполнение курсового проекта способствует:

- формированию у студентов умений в реализации основных методик, моделей и технологий разработки приложений в соответствии с современными требованиями и мировыми стандартами программной инженерии;
- углублению теоретических знаний в соответствии с заданной темой;
- разработка студентом программного обеспечения, отвечающего требованиям к уровню подготовки специалиста по специальности 09.02.07 «Информационные системы и программирование» квалификации Программист.
- формированию умения использовать справочную, нормативную и правовую документацию;
- развитию творческой инициативы, самостоятельности, ответственности и организованности;
- подготовке к итоговой государственной аттестации.

Курсовой проект является важнейшей формой самостоятельной работы. В процессе его выполнения студенты приобретают умения и навыки проведения научных исследований.

Выполненная студентом курсовая работа проверяется преподавателем-руководителем, который составляет по ней рецензию-отзыв. Законченная работа проходит процедуру защиты. Курсовой проект защищается до сдачи экзаменов.

2 Основные этапы выполнения курсового проекта

Тематика курсового проектирования определяется программой МДК 02.01 «Технология разработки программного обеспечения».

При выборе темы необходимо учитывать, в какой мере разрабатываемые вопросы актуальны для работодателя, обеспечены исходными данными, литературными источниками, соответствуют индивидуальным способностям и интересам студента.

После того как тема курсовой работы выбрана и согласована с руководителем (преподавателем), оформляется бланк задания (см. Приложение 1) и составляется календарный план (см. Приложение 2), в котором определяются сроки выполнения этапов курсовой работы. План облегчает контроль за ходом выполнения исследования и помогает студенту самостоятельно и осознано выполнять курсовую работу.

Учитывая специфику МДК 02.01 «Технология разработки программного обеспечения», процесс выполнения курсового проекта включает в себя ряд взаимосвязанных этапов:

1. Провести анализ предметной области.
2. Составить техническое задание или технические требования к программному обеспечению (ПО).
3. Выбрать модель жизненного цикла (ЖЦ) разработки ПО, обосновав выбор модели ЖЦ для данной разработки и описав использованную модель применительно к данной разработке.
4. Спроектировать программное обеспечение.
5. Реализовать проект с помощью выбранного языка программирования или программного средства.
6. Выполнить тестирование и отладку ПО.
7. Выполнить экономическую оценку программного продукта.
8. Описать, оформить и сдать на проверку курсовой проект: электронную версию программного обеспечения и пояснительную записку.

3 Требования к содержанию курсового проекта

- По содержанию курсовой проект по МДК 02.01 «Технология разработки программного обеспечения» носит технологический характер.
- По структуре курсовой проект состоит из пояснительной записки и практической части – электронной версии разработанного программного обеспечения.

Все этапы курсового проектирования должны быть отражены в пояснительной записке.

Пояснительная записка курсового проекта технологического характера содержит следующие общие разделы:

Титульный лист

Содержание

Введение

1. Анализ предметной области
2. Разработка технического задания или технических требований к программному обеспечению (ПО)
3. Выбор модели жизненного цикла ПО
4. Проектирование ПО
5. Реализация спроектированного программного обеспечения
6. Выполнение тестирования и отладки ПО
7. Экономическая оценка ПО

Заключение

Список использованной литературы

Приложения

Образец титульного листа и содержания приводятся в Приложениях 3 и 4.

Содержание

В содержании последовательно излагаются наименования глав, разделов и подразделов курсового проекта. При этом их формулировки

должны точно соответствовать содержанию работы, быть краткими, четкими, последовательно и точно отражать её внутреннюю логику.

В содержании указывают страницы, с которых начинаются каждая глава, раздел или подраздел. Страницы в работе должны быть пронумерованы. Счет нумерации страниц начинается с титульного листа, на котором номер страницы не указывается. Введение, отдельные главы, заключение, список литературы и каждое приложение **должны** всегда **начинаться на новой странице**. Пример оформления содержания приводится в Приложении 4. Текст работы должен соответствовать содержанию.

Введение

Во введении анализируются условия задачи, приводится обзор литературы по теме задания, дается анализ информации с точки зрения приложения к поставленной задаче.

Введение – это обоснование и доказательство важности рассматриваемой темы. Введение знакомит с существом рассматриваемого вопроса, вводит в тему.

Введение к курсовой работе в обязательном порядке содержит следующие элементы:

- **Определение темы работы.** Необходимо привести несколько (2–3) фраз из литературы, характеризующих основные понятия темы. Например, для темы «Технология разработки приложения для предметной области «Экономика» с применением языка программирования С#»: **«Язык программирования»** - формальная знаковая система, предназначенная для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, задающих внешний вид программы и действия, которые выполнит исполнитель (компьютер) под её управлением.

- **Актуальность** работы. Следует обозначить существующее положение, почему именно это проблема актуальна. Обоснование может начинаться с фразы «Актуальность темы исследования обусловлена тем, что

создание программного обеспечения на данном этапе развития информационных технологий...» или «Данная тема актуальна, так как ...».

- **Цель** работы. Цель показывает направление раскрытия темы работы. Выглядеть это может следующим образом: *«Цель курсового проекта - исследовать возможность создания приложений по данной предметной области в среде программирования С#».*

Или «Целью данной работы является изучение (описание, определение, установление, исследование, рассмотрение, разработка, раскрытие, освещение, выявление, анализ, обобщение....

Необходимо отметить важное правило – введение, как и заключение, рекомендуется писать после полного завершения основной работы. До того, как будет создана основная часть работы, реально невозможно написать хорошее введение, так как автор еще не вполне овладел материалами по теме.

Не рекомендуется делать объем введения более 2-2,5 страниц.

Рассмотрим пример возможных задач курсового проекта.

В данной работе будет рассмотрена технология разработки информационной системы (ИС) «Гостиница».

Задачи, которые ставятся в проекте:

- *провести анализ предметной области – деятельности персонала гостиницы – объекта проекта;*
- *определить технические требования к программному обеспечению (ПО);*
- *подобрать модель жизненного цикла ПО;*
- *выполнить проектирование базы данных – ядра ИС и программы-оболочки;*
- *реализовать проект с учетом выбранных программных средств;*
- *провести тестирование и отладку ПО;*
- *выполнить экономическую оценку созданного приложения.*

Анализ предметной области

В разделе «Анализ предметной области» указывают назначение разрабатываемого программного продукта и дают краткую характеристику области его применения, например:

В данном курсовом проекте предметной областью для решения поставленных задач является гостиница. Это предприятие предназначено для организации предоставления комнат посетителям на предмет проживания их на различные сроки пребывания. Комнаты в гостинице, предоставляемые для проживания, подразделяются на категории по комфортности номеров: «люкс» и обычные и по количеству койко-мест - одноместные, двухместные, трехместные. Всего в гостинице 100 мест.

Обслуживание постояльцев организуется следующим образом:

- *Прибывающий в гостиницу (в дальнейшем - постоялец) сообщает о себе сведения администратору путем заполнения карточки учета постояльцев гостиницы: фамилию, имя, отчество, паспортные данные (серию, номер, кем и когда выдан, по какому адресу прописан), срок пребывания в данном населенном пункте, желаемый номер («люкс» или обычный и - одноместный, двухместный и т.д.).*
- *Администратор анализирует наличие свободных мест в гостинице и выдает ключ от номера постояльцу после оплаты им предоставляемых услуг в кассе.*

Возникает необходимость в компьютерной обработке вводимых данных о постояльцах с целью хранения этой информации на носителях, а также для анализа заполняемости гостиницы, контроля свободных мест, свободных и занятых номеров.

Организуется обследование предметной области: оценка объема и цели проекта, определение требований, объектов и функций на высоком уровне.

Сбор информации начинается с изучения существующих форм документов, отчетов, имеющихся файлов, баз данных, программ.

Исходная информация для анализа берется из индивидуальных бесед с заказчиками, на семинарах, при изучении документации, инструкций, анкетировании и др.

Анализ должен заканчиваться подробным описанием информации об объектах предметной области, которая требуется для решения конкретных задач и которая должна храниться в БД, формулировкой конкретных задач, которые будут решаться с использованием данной БД с кратким описанием алгоритмов их решения, описанием выходных документов, которые должны генерироваться в системе, описанием входных документов, которые служат основанием для заполнения данными БД.

Анализ предметной области может быть продемонстрирован диаграммами моделирования, разработанными с применением CASE-средств.

Разработка технического задания (или требований к программному обеспечению)

В данном разделе составляется техническое задание на разработку ПО.

Разработка технических требований к программному продукту является составной частью стадии технического задания (предпроектной стадии), почти полностью совпадающей с этапом системного анализа. Этапы разработки технического задания следующие:

- сбор сведений (обследование);
- обработка сведений обследования и подготовка технико-экономического обоснования;
- техническое задание (завершающий этап).

Требования к функциональным характеристикам (пример)

Разрабатываемое ПО должно обеспечивать:

- *сбор и анализ информации о продуктах ресторана: наименование товарных групп (мясо-рыбные продукты, молочные, зерно-мучные, плодоовощные, вкусовые, крахмал-сахар) и наименования продуктов, составляющих эти товарные группы, а также их вес/количество, цена единицы, общая стоимость продукта;*
- *выдачу рекомендаций по дальнейшей работе; отображение текущего*

состояния по набору параметров - циклически постоянно (режим работы круглосуточный),

- *визуализацию информации о продуктах, имеющихся на складе ресторана.*

Организация входных и выходных данных

Исходные данные в систему поступают с помощью ввода оператором данных о поступающих товарах. После анализа поступившей информации они сохраняются в базе данных «Ресторан».

Выбор модели жизненного цикла ПО

Необходимо выбрать модель жизненного цикла (ЖЦ) разработки программного обеспечения, обосновав выбор модели ЖЦ для данной разработки и описав использованную модель применительно к данному проекту.

Например, рассмотрим следующий возможный выбор модели жизненного цикла.

Для разрабатываемого программного обеспечения (ПО) базы данных «Ресторан» подходит модель ЖЦ, называемая «Прототипирование», которая позволяет создать прототип ПО до или в течение этапа составления требований ПП.

В результате прототипирования разработчик демонстрирует пользователям готовый прототип, а пользователи оценивают его функционирование.

Модель прототипирования обладает целым рядом преимуществ:

- *взаимодействие заказчика с разрабатываемой системой начинается на раннем этапе;*
- *благодаря реакции заказчика на прототип, сводится к минимуму число неточностей в требованиях;*
- *...*
- *заказчики принимают участие в процессе разработки на протяжении всего жизненного цикла и в конечном итоге в большей степени довольны результатами работы.*

Модель прототипирования рекомендуется применять в следующих случаях:

- требования к ПО заранее неизвестны;
- ...
- выполняется новая, не имеющая аналогов разработка;
- разработчики не уверены в том, какое решение следует выбрать.

Поэтому с уверенностью можно сказать, что данная модель жизненного цикла программного продукта соответствует жизненному циклу информационной системы «Ресторан».



Рис. 1 Модель прототипирования для ИС «Ресторан»

Проектирование программного обеспечения

Далее выполняется проектирование программного обеспечения. Выбирается и обосновывается метод решения задачи, например:

В процессе проектирования (design) основное внимание уделяется логическому решению, обеспечивающему выполнение основных требований. В реальной жизни программные проекты чаще всего достаточно сложны, и их декомпозиция (по принципу "разделяй и властвуй") - это основная и, наверное, единственная стратегия борьбы со сложностью. Она состоит в разбиении проблемы на мелкие управляемые

элементы. Существуют и стандартизированные методологии и инструментальные средства, прошедшие проверку временем, позволяющие решить эту задачу.

Рассмотрим построения инфологической модели базы данных "Ресторан", где должна храниться информация о блюдах, их ежедневном потреблении, продуктах, из которых приготавливаются эти блюда, и поставщиках этих продуктов. Информация будет использоваться поваром и руководителем небольшого предприятия общественного питания, а также его посетителями.

Выделим следующие объекты и характеристики проектируемой базы:

- Блюда, для описания которых нужны данные, входящие в их кулинарные рецепты: номер блюда (например, по кулинарной книге) название блюда, вид блюда (закуска, суп, горячее и т.п.), выход (вес порции), калорийность*
- Для каждого поставщика продуктов: наименование, адрес, название поставляемого продукта, дата поставки и цена на момент поставки.*
- Ежедневное потребление блюд (расход): блюдо, количество порций, дата.*

Анализ объектов позволяет выделить:

- стержни Блюда, Продукты и Города;*
- ассоциации Состав (связывает Блюда с Продуктами)*

Поставки (связывает Поставщиков с Продуктами);

- обозначение Поставщики;*
- характеристики Рецепты и Расход.*

Разработаем ER-диаграмму модели, а также модель на языке инфологического моделирования (ЯИМ).

Модель на ЯИМ имеет следующий вид:

Блюда (БЛ, Блюдо, Вид)

Продукты (ПР, Продукт, Калорийность)

Поставщики (ПОС, Город, Поставщик) [Город]
 Состав [Блюда М, Продукты N] (БЛ, ПР, Вес (г))
 Поставки [Поставщики М, Продукты N] (ПОС, ПР, Дата_П, Цена, Вес (кг))
 Города (Город, Страна)
 Рецепты (БЛ, Рецепт) {Блюда}
 Расход (БЛ, Дата_Р, Порций) {Блюда}

В этих моделях Блюдо, Продукт и Поставщик – наименования, а БЛ, ПР и ПОС – цифровые коды блюд, продуктов и организаций, поставляющих эти продукты.

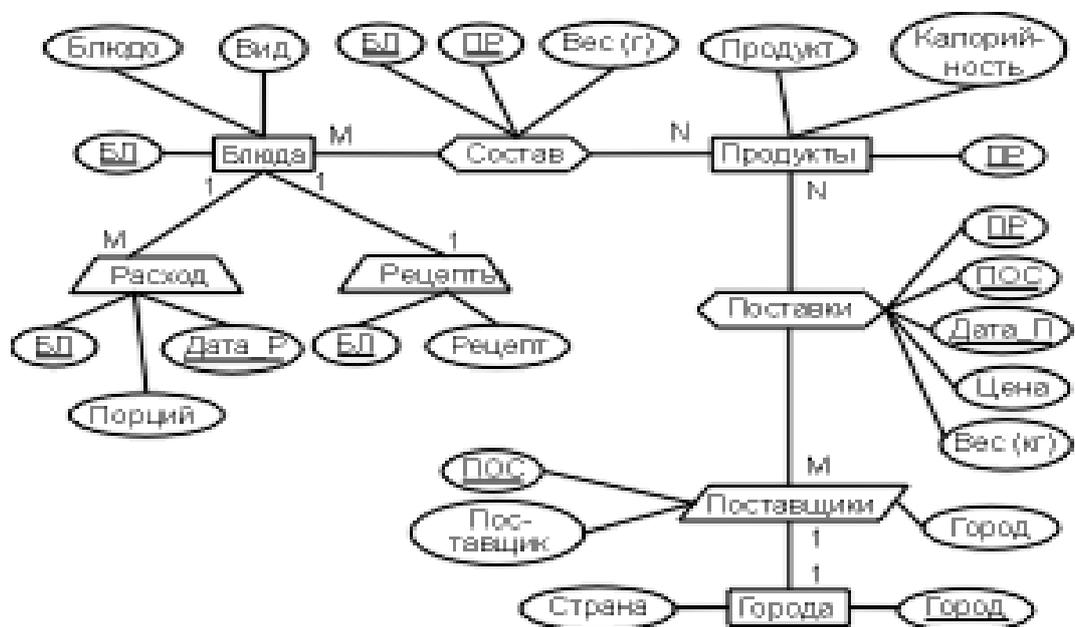


Рис. 2 Инфологическая модель базы данных "Ресторан"

Реализация программного обеспечения с помощью языка программирования (или программного средства)

На основе полученных результатов делается выбор программного обеспечения для решения поставленной задачи.

Следующим этапом выполнения КП является разработка алгоритма решения задачи. В зависимости от типа решаемой задачи уровень детализации схемы алгоритма может быть различным. Часто используется метод пошаговой детализации. Метод пошаговой детализации предполагает

поэтапную разработку алгоритма. На первом этапе строится обобщенный алгоритм, состоящий из укрупненных блоков, затем происходит постепенное уточнение последовательности действий, выполняемых в каждом блоке. Этапы разработки алгоритма должны быть отражены в пояснительной записке.

В случае если для решения задачи используется язык программирования, пояснительная записка должна содержать описание этапов разработки программы, листинг программы.

Рассмотрим пример описания этапов разработки программы.

Для спроектированного программного продукта – базы данных «Ресторан» выберем систему программирования АВСПаскаль, которая позволяет программировать базы данных и организовывать данные в файлах на носителях информации – магнитных дисках, а также обладает следующими свойствами: ориентация на работу с мышью, использование многооконного режима работы, объектно-ориентированная библиотека и возможность включать в текст программ команды Ассемблера.

В АВСПаскале информацию баз данных можно представить в виде структурированного типа данных, называемом – запись. Запись – это структурированный тип, состоящий из фиксированного числа компонент (полей базы данных) разного типа. Обычно запись содержит совокупность разнотипных атрибутов, относящихся к одному объекту. Сведения о ресторане можно представить в виде таблицы базы данных.

Таблица 1 - Примерный вид таблицы базы данных «Ресторан»

<i>Номер блюда</i>	<i>Название блюда</i>	<i>Вид</i>	<i>Выход</i>	<i>Калорийность</i>
1	Лобио	Закуска	230	3010
2	Харчо	Суп	310	1500
3	Шашлык	Горячее	180	2000
4	Кофе	Десерт	200	2750

Такая структура называется двухуровневым деревом. Эта информация будет храниться в переменной типа *Record* (запись).

Зададим *тип bl*:

Type bl = record

nombl: integer;

nazbl: string [20];

vid: string[10];

vih: integer;

kalor: integer;

end;

Затем опишем соответствующую переменную следующим образом
var restor: array[1..40] of bl ;

Переменная *restor* объявляется массивом из 40 элементов типа *bl*.

Здесь в *типе bl*, который является записью, объявлены следующие элементы записи:

- *nombl* – номер блюда (например, по кулинарной книге);
- *nazbl* – название блюда;
- *vid* – вид блюда (закуска, суп, горячее и т.п.);
- *vih* – выход (вес порции);
- *kalor* – калорийность,

Любая обработка записей производится поэлементно. Ввод информации о блюдах ресторана можно организовать следующим образом:

For i:=1 to 40 do

Begin

write ('введите номер блюда');

readln (restor [i]. nombl);

write ('введите название блюда');

readln (restor [i]. nazbl);

write ('введите вид блюда');

readln (restor [i]. vid);

Write ('введите выход блюда');

readln (restor [i]. vih);

Write ('Какая калорийность блюда);

readln (restor [i]. kalor) end;

Здесь используются двойные составные имена переменных, предназначенные для организации ввода данных о элементах информационной системы «Ресторан», например restor [i]. Nazbl и др.

Примечание. При использовании другого программного средства пояснительная записка должна содержать правила работы с пакетом программ и точную последовательность выполняемых действий при решении задачи.

Более полную информацию по содержанию этого раздела студент получает из специальной методической литературы или на консультации у руководителя КП.

Тестирование и отладка программного обеспечения

В данном разделе описывается тестирование и отладка разработанного программного обеспечения. Например, для программы, написанной на ABC Паскале процесс отладки может иметь следующее описание:

Турбо-система помогает обнаружить ошибку в программе, если при исполнении произошел сбой. Пользователю сообщается причина сбоя и указывается место, где он случился в Паскаль-программе. Также имеется система отладки Debug, с помощью которой можно просмотреть на экране значение любой переменной, найти значение любого выражения, установить новое значение переменной. Можно прервать выполнение программы в указанных местах, которые называются контрольными точками. Система отладки существенно облегчает программисту поиск ошибок.

Затем приводятся примеры отладки на примере обнаружении ошибок и способов их устранения.

Экономическая оценка программного обеспечения

Необходимо выполнить экономическую оценку программного продукта, например, рассчитать отпускную цену ПО и цену с учетом предложения по прибыли.

При формировании договорной цены сначала определяют предварительную договорную цену (ПДЦ), которая складывается из составляющих:

$$ПДЦ = C + П + Н$$

C – себестоимость;

$П$ – прибыль (25% от себестоимости);

$Н$ – налоги (НДС 18% от суммы себестоимости и прибыли), т.е.

$$П = 0,25 * C;$$

$$Н = 0,18 * (C + П);$$

Другими словами - это свободная отпускная цена производителя СОЦП.

После того как подготовлено предложение о ПДЦ возможна и корректировка:

$$ДЦ = ПДЦ + ПП$$

Здесь ПП - предложения по прибыли.

Аналогично определяются цены на услуги по сопровождению программной продукции, с той лишь разницей, что цена на услуги может быть связана не с самим фактором проведения работ, а с периодом в течение которого оказываются эти услуги.

Рассчитаем отпускную цену производителя программного продукта. Примем за начало расчета себестоимость $C = 210000$ р., средняя себестоимость программного продукта, произведенного группой разработчиков. По процентному соотношению программный продукт – база данных «Ресторан» составляет 10% от принятой себестоимости, т.е. наша расчетная себестоимость $C = 21000$ р.

$$\text{Рассчитаем прибыль } П = 0,25 * 21000 = 5250 \text{ р.}$$

$$\text{Тогда можно подсчитать налоги } Н = 0,18 * (21000 + 5250) = 0,18 * 26250 = 4725 \text{ р.}$$

Отсюда следует, что отпускная цена производителя вычисляется следующим образом:

$$ПДЦ = C + П + Н;$$

$$ПДЦ = 21000 + 5250 + 4725 = 30975 \text{ р.}$$

Также можно определить договорную цену программного продукта с учетом предложения по прибыли, например 30% от ПДЦ.

$$ДЦ = ПДЦ + ПП:$$

$$ДЦ = 30975 + 30975 * 0,3 = 30975 + 9292,5 = 40267,5 \text{ р.}$$

Заключение

Заключение содержит краткую характеристику результатов, полученных в ходе работы.

В заключении автор курсового проекта анализирует полученные результаты, делает выводы, обобщения и рекомендации, которые вытекают из его работы, подчеркивает их практическую значимость.

Необходимо иметь в виду, что введение и заключение никогда не делятся на части.

Объем заключения примерно равен объему введения.

Библиография (список использованной литературы)

Библиография - это перечень книг, журналов, статей с указанием основных данных (место и год выхода, издательство и др.).

Библиографический список источников и литературы, использованных при разработке КП, является обязательным атрибутом любой научно-исследовательской работы, в том числе курсовой.

Библиографический список включает библиографические описания всех использованных, цитированных или упоминаемых в работе документов, а также прочитанной литературы по теме, которая оказала существенной влияние на содержание работы.

Приложения

Обязательным компонентом КП является приложение в виде файла с электронным вариантом решения задачи, которая сохраняется в электронном виде на носителе информации вместе с пояснительной запиской.

Также в качестве приложений могут присутствовать таблицы, схемы, графики, диаграммы, скрины экрана и другие документы, поясняющие процесс разработки решения задачи.

Приложения оформляются после списка источников и литературы и располагаются в порядке ссылок в тексте. Каждое приложение начинается с нового листа с обозначением в правом верхнем углу словом «Приложение». Приложения должны нумероваться последовательно, арабскими цифрами (например, «Приложение 10») и иметь заголовок. Если приложение одно, то оно не нумеруется.

В приложении в левом углу должно быть указано, на основании каких источников оно составлено.

4 Оформление курсового проекта

4.1 Общие требования

Тема курсового проекта должна раскрываться в вопросах. Количество вопросов в курсовом проекте строго не регламентируется. Однако следует учитывать, что в работе, как правило, рассматриваются теоретические и практические аспекты исследуемой темы.

Работа готовится в одном экземпляре.

Текст выполняется в текстовом процессоре Microsoft Word на персональном компьютере.

Общий объем курсового проекта - не менее 20 страниц печатного текста и обычно не превышает 25 страниц.

В данный объем не включаются: приложения, список литературы.

Работа выполняется на белой бумаге формата А4 (210 x 297 мм). Текст работы излагается на одной стороне листа. Каждая страница работы оформляется следующими полями: левое – 30 мм, правое – 10 мм, верхнее – 15 мм, нижнее – 20 мм. Абзац оформляется отступом красной строки не более 15 мм.

Все страницы нумеруются арабскими цифрами по порядку от титульного листа до последнего (списка литературы) в середине верхнего поля. Первой страницей считается титульный лист, на котором номер страницы не ставится.

Нумерация вопросов и подвопросов осуществляется арабскими цифрами без точки в конце номера.

Содержание работы помещается после задания на курсовой проект. Слово «содержание» записывают в виде заголовка симметрично тексту прописными (заглавными) буквами.

Иллюстрации (изображения окон программ, рисунки, схемы, графики, диаграммы и др.), за исключением иллюстраций приложений, следует нумеровать арабскими цифрами сквозной нумерацией отдельно по каждому виду иллюстраций (например, Рис. 1 – Изображение окна программы).

Обычно они размещаются непосредственно после первого упоминания о них, либо сверху следующей страницы.

4.2 Ссылки

Ссылка является указанием источника, на который ссылаются. Ссылка в тексте приводится в виде номера, соответствующего литературному источнику или нормативному документу, приведенному в библиографическом списке, и заключенного в скобки.

При оформлении сносок их помещают на той же странице под строками основного текста с отделением от него небольшой горизонтальной линией.

Примечание. Ссылки на использованную литературу обязательны, так как в этом проявляется культура отношения к чужой мысли, чужому тексту.

4.3. Основные элементы библиографического описания

Библиографическое описание документов, включенных в библиографический список, составляется в соответствии с требованиями ГОСТ 7.1-2003. «Библиографическое описание документа. Общие требования и правила составления».

Развернутая методика библиографического описания изложена в «Правилах составления библиографического описания».

Последовательность формирования библиографического списка может быть различной:

- по значимости документов (нормативные акты, документальные источники, монографии, статьи, прочая литература);
- по алфавиту фамилий авторов или названий документов;
- по хронологии издания документов и т.п.

Основные элементы библиографического описания приводятся в следующей последовательности (ГОСТ 7.0-84 «Библиографическая деятельность. Термины и определения»):

- фамилия автора и его инициалы;
- название книги без кавычек;
- место издания;

- название издательства;
- год издания;
- номер (номера) страницы.

Примеры библиографического описания:

Литература:

1. Федорова Г. Н. Разработка модулей программного обеспечения для компьютерных систем: учебник для студ. Учреждений сред. проф. образования/ Г. Н. Федорова – М.: Издательский центр «Академия»; 2017. – 384 с.
2. Рудаков А.В. Технология разработки программных модулей: учеб. для студ. учреждений сред. проф. образования/А.В. Рудаков. - 12-е изд. - М.: Издательский центр «Академия», 2018. - 208 с.

4.4. Правила сокращения слов и словосочетаний

Применение сокращенных словосочетаний регламентируется ГОСТ 7.12-93 «Сокращение русских слов и словосочетаний в библиографическом описании».

Кроме того, имеются общепринятые правила сокращения слов и выражений, применяемые при написании курсовых работ, рефератов, диссертаций, статей. При этом используются следующие способы:

- Пишут лишь первые буквы слова (например, «гл.» - глава, «ст.» - статья).
- Оставляют лишь первую букву слова (например «в» - век, «г» - год).
- Оставляют только часть слова без окончания и суффикса (например, «абз» - абзац).
- Пропускают сразу несколько букв в середине слова, а вместо них ставят дефис (например, университет – «ун-т», издательство – «изд-во»).

Студенты должны быть также внимательны при использовании и таких трех видов сокращений, как буквенные аббревиатуры, сложносокращенные слова, условные сокращения по начальным буквам слов или по частям слов.

Таковыми аббревиатурами удобно пользоваться, так как они составляются из общеизвестных словообразований (например, «ПЗУ», «ОС»). Если необходимо обозначить свой сложный термин такой аббревиатурой, то в этом случае ее следует указывать сразу же после данного сложного термина. Например, «Электронные таблицы (ЭТ)». Далее этой аббревиатурой можно пользоваться без расшифровки.

При написании курсовой и дипломной работы студент должен соблюдать общепринятые графические сокращения по начальным буквам слов или по частям таких слов: «и т.д.» (и так далее), «и т.п.» (и тому подобное), «и д.р.» (и другое), «т.е.» (то есть), «и пр.» (и прочее), «вв.» (века), «гг.» (года), «н.э.» (нашей эры), «обл.» (область), «гр.» (гражданин), «доц.» (доцент), «акад.» (академик). При сносках и ссылках на источники употребляются такие сокращения, как «ст.ст» (статьи), «см.» (смотри), «ср.» (сравни), «напр.» (например), «т.т.» (тома).

Следует иметь также в виду, что внутри самих предложений такие слова, как «и другие», «и тому подобное», «и прочее» не принято сокращать. Не допускаются сокращения слов «так называемый» (т.н.), «так как» (т.к.), «например» (напр.), «около» (ок.), «формула» (ф-ла).

5. Аттестация курсового проекта

Аттестация курсового проекта осуществляется по 4-бальной системе с обязательным оформлением «Рецензии-отзыва курсового проекта»

Критерии оценки курсового проекта, посвященного разработке программного продукта:

- **Отличная** оценка выставляется за курсовые проекты, которые имеют конкретный практический материал по технологии разработки программного обеспечения, содержат элементы научного поиска, грамотно и аккуратно оформлены в соответствии с требованиями, выполнены и защищены в срок.
- **Хорошая** оценка выставляется при невыполнении студентом одного из вышеперечисленных требований, но при соблюдении сроков подготовки.
- **Удовлетворительно** оцениваются компилятивные (чисто описательные) работы при условии, что соблюдены сроки написания и требования к ее оформлению.
- **Неудовлетворительная** оценка ставится в том случае, если студент не представил (по неуважительной причине) готовый курсовой проект к сроку защиты.

ПРИЛОЖЕНИЕ 1

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
«ГЕОРГИЕВСКИЙ РЕГИОНАЛЬНЫЙ КОЛЛЕДЖ «ИНТЕГРАЛ»

Отделение: Индустриальных и цифровых технологий
ПЦК Информационных технологий и программирования

ЗАДАНИЕ

на курсовой проект

«Технология разработки информационной системы «Автобаза»
по МДК 02.01. «Технология разработки программного обеспечения»
для студента IV курса группы ИСП-91

Иванова Ивана Ивановича

1. Разработать информационную систему для автобазы
2. Составить пояснительную записку следующего содержания:

Введение

1 Анализ предметной области

2 Разработка технического задания

3 Выбор модели жизненного цикла программного обеспечения

4 Проектирование информационной системы

5 Реализация проекта с помощью выбранного программного средства

6 Тестирование и отладка программного обеспечения

7 Экономическая оценка программного обеспечения

Заключение

Список используемой литературы

Приложение

3. Объем проекта: 25 стр.

4. Срок проектирования: 28 февраля 2023 г.

Руководитель проекта _____ А.А. Петрова

Задание к выполнению принял студент _____ И.И. Иванов

Дата выдачи задания 16 января 2023 г.

ПРИЛОЖЕНИЕ 2

График выполнения курсового проекта по МДК 03.01 «Технология разработки программного обеспечения»

Раздел КП	Даты выполнения	Примечание
Анализ предметной области	16.01 – 15.01.2023	
Техническое задание	18.01 – 22.01	
Выбор модели жизненного цикла ПП	25.01 – 29.01	
Проектирование программного обеспечения	1.02 – 5.02	
Реализация программного обеспечения	8.02 – 12.02	
Тестирование и отладка программного обеспечения Составление пояснительной записки	15.02 – 21.02	
Экономическая оценка ПО Написание Введения и заключения	21.02 – 24.02	
Окончательное оформление пояснительной записки	24.02 – 28.02	

ПРИЛОЖЕНИЕ 3

Образец оформления титульного листа курсового проекта

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СТАВРОПОЛЬСКОГО КРАЯ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
«ГЕОРГИЕВСКИЙ РЕГИОНАЛЬНЫЙ КОЛЛЕДЖ «ИНТЕГРАЛ»

Отделение: Индустриальных и цифровых технологий
ПЦК Информационных технологий и программирования
Специальность 09.02.07 Информационные системы и программирование

Пояснительная записка

Курсовой проект

по МДК 02.01. «Технология разработки программного обеспечения»

Тема работы «Технология разработки ВАША ТЕМА

»

Выполнил студент **ФИО**

Курс **IV**

Группа **ИСП-91**

Работа выполнена "___" _____ 2023 г. _____

(подпись выпускника)

Защитил работу с оценкой _____

Руководитель _____ **А.А. Петрова**

"___" _____ 2023 г.

ГЕОРГИЕВСК

2023

ПРИЛОЖЕНИЕ 4

Образец Содержания



СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	4
1.1. Характеристика кадрового агентства.....	4
2 РАЗРАБОТКА ТЕХНИЧЕСКОГО ЗАДАНИЯ НА РАЗРАБОТКУ ИНФОРМАЦИОННОЙ СИСТЕМЫ	5
3 ВЫБОР МОДЕЛИ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	6
4 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА.....	7
4.1 Основы проектирования информационной системы (ИС).....	7
4.2 Моделирование ИС.....	7
5 РЕАЛИЗАЦИЯ ПРОЕКТА С ПОМОЩЬЮ ВЫБРАННОГО ПРОГРАММНОГО СРЕДСТВА	8
5.1 Выбор инструментария для реализации проекта	8
5.2 Разработка базы данных.....	8
5.3 Реализация интерфейса системы.....	8
6 ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	9
7 ЭКОНОМИЧЕСКАЯ ОЦЕНКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	10
ЗАКЛЮЧЕНИЕ	11
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ ИСТОЧНИКОВ.....	12
ПРИЛОЖЕНИЯ.....	13

					КП – 09.02.07 – ваш номер – 2023 - ПЗ			
Изм.	Лист	№ докум.	Подп.	Дата	Технология разработки ВАША ТЕМА	Лит.	Лист	Листов
Разраб.		ФНО				У	2	
Пров.		Петрова А.А.				ГБПОУ ГРК «Интеграл» ИСП-91		
Н.контр.								
Утв.								