



ДИПЛОМНОЕ ПРОЕКТИРОВАНИЕ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

МИНСК 2008

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«МИНСКИЙ ГОСУДАРСТВЕННЫЙ ВЫСШИЙ
РАДИОТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

ДИПЛОМНОЕ ПРОЕКТИРОВАНИЕ

Методические указания
для учащихся специальности 2-40 01 01
«Программное обеспечение информационных технологий»

МИНСК 2008

УДК 681.3.02(075)
ББК 32.973я7
Д46

Рекомендовано к изданию кафедрой информатики и Научно-методическим советом Учреждения образования «Минский государственный высший радиотехнический колледж» (протокол № 3 от 14.11.07 г.)

С о с т а в и т е л и :

М. А. Бельчик, ассистент кафедры информатики МГВРК
М. В. Василевская, ассистент кафедры информатики МГВРК
Ю. А. Скудняков, зав. кафедрой информатики МГВРК,
канд. техн. наук, доцент
И. М. Снежкова, старший преподаватель кафедры
информатики МГВРК

Р е ц е н з е н т

Л. Н. Громова, старший преподаватель
кафедры информатики МГВРК

Д46 **Дипломное проектирование** : метод. указания для учащихся специальности 2-40 01 01 «Программное обеспечение информационных технологий» / сост. М. А. Бельчик, М. В. Василевская, Ю. А. Скудняков, И. М. Снежкова. – Мн. : МГВРК, 2008. – 40 с.
ISBN 978-985-6851-26-4

Настоящие методические указания предназначены для организации дипломного проектирования по специальности 2-40 01 01 «Программное обеспечение информационных технологий» и включают указания по структуре и содержанию разделов дипломного проекта.

Предназначено для учащихся и преподавателей колледжа.

УДК 681.3.02(075)
ББК 32.973я7

ISBN 978-985-6851-26-4

- © Бельчик М. А., Василевская М. В.,
Скудняков Ю. А., Снежкова И. М.,
составление, 2008
© Оформление. Учреждение образования «Минский государственный высший радиотехнический колледж», 2008

Предисловие

В условиях глобальной информатизации общества неотъемлемым качеством специалиста является высокий уровень информационной культуры. Это предполагает знание особенностей информационных потоков в разных предметных областях, умение применять современные информационные технологии для автоматизации ручных работ и решения широкого круга нестандартных задач.

Необходимость информатизации всех областей функционирования общества диктует такие условия подготовки специалистов, чтобы они стали разработчиками, создателями и пользователями информационных систем, владели информационными технологиями, умели строить и оценивать работу информационных систем, качество обработки, полноту и точность информации в сфере профессиональной и социальной деятельности информационного общества.

Специалистами такого профиля являются выпускники специальности 2-40 01 01 «Программное обеспечение информационных технологий», а дипломное проектирование является формой итогового контроля знаний учащихся.

Дипломное проектирование в колледже организуется в соответствии с «Положением о текущей и итоговой аттестации учащихся учреждений, обеспечивающих получение среднего специального образования», утвержденным Министерством образования Республики Беларусь 15.10.2002 г.

Дипломное проектирование в соответствии с учебным планом осуществляется в сроки с 4 мая по 21 июня, защита – с 22 июня.

Организация, общее руководство, контроль за ходом дипломного проектирования осуществляются кафедрой информатики.

Руководство экономическим разделом осуществляется кафедрой экономики и управления.

Руководство разделом охраны труда осуществляется кафедрой общетехнических дисциплин.

1 ЦЕЛИ ДИПЛОМНОГО ПРОЕКТИРОВАНИЯ

Целями дипломного проектирования являются:

- систематизация, закрепление и углубление теоретических знаний и практических умений и навыков учащихся специальности 2-40 01 01 «Программное обеспечение информационных технологий»;
- применение полученных знаний для решения конкретных задач;
- совершенствование навыков ведения самостоятельной проектной работы и реализации проекта в производстве;
- совершенствование приемов и стиля разработки программного обеспечения в различных приложениях.

Дипломный проект является квалификационной работой выпускника. По уровню выполнения дипломного проекта и результатам его защиты Государственной квалификационной комиссией (ГКК) делается заключение о возможности присвоения выпускнику квалификации техник-программист.

2 ОРГАНИЗАЦИЯ ДИПЛОМНОГО ПРОЕКТИРОВАНИЯ

Написание дипломного проекта (ДП) – процесс достаточно сложный и трудоемкий, включающий разработку темы, экономические расчеты, составление мероприятий по охране труда и технике безопасности.

Тематика дипломных проектов должна быть актуальной, соответствовать современному состоянию и перспективам развития информационных технологий в Республике Беларусь. Тематика дипломных проектов разрабатывается преподавателями кафедры информатики, рассматривается и корректируется на заседании кафедры. Утверждение тем дипломных проектов учащихся осуществляется по их письменному заявлению и оформляется приказом ректора перед направлением на преддипломную практику. Заявление согласовывается с руководителем диплома и заведующим кафедрой. Тема дипломного проекта назначается, как правило, индивидуально. Для крупных тем может быть привлечено несколько дипломников с четким разграничением решаемых задач.

Руководитель дипломного проекта разрабатывает индивидуальное задание и график выполнения проекта для каждого уча-

щегося. Задания рассматриваются кафедрой информатики, утверждаются заведующим кафедрой и выдаются учащимся до начала преддипломной практики.

Руководитель проекта:

- оказывает помощь в подборе материалов и литературы для выполнения проекта;
- проводит консультации и обеспечивает своевременное и качественное выполнение ДП;
- подготавливает отзыв о ДП.

Контроль за ходом выполнения ДП осуществляет кафедра информатики. Рабочие комиссии, формируемые из преподавателей, осуществляют технический контроль и нормоконтроль.

Первая проверка (опроцентовка) готовности проекта проводится примерно через три недели после начала дипломного проектирования. Требуемый объем выполняемого проекта должен быть не менее 30 % от всего объема. Вторая проверка осуществляется с интервалом в две недели, при этом должно быть выполнено 90 % всего объема ДП.

Заключительная проверка осуществляется в виде рабочей комиссии (предварительной защиты). На рабочую комиссию учащийся должен представить:

- дипломный проект в полном объеме, что означает наличие пояснительной записки с титульным листом, подписанным дипломником, руководителем и всеми консультантами; наличие графической части, подписанной руководителем и дипломником;
- отзыв руководителя.

Во время рабочей комиссии осуществляются проверка готовности дипломника к защите и нормоконтроль. Рабочая комиссия заслушивает доклад учащегося, подготовленный к защите, проверяет соответствие оформления пояснительной записки и графической части стандартам и принимает решение о возможности допуска дипломника к защите.

По представлению рабочей комиссии и отзыву руководителя заведующий кафедрой информатики назначает рецензента проекта. После получения рецензии допуск учащегося к защите фиксируется подписью заведующего отделением на титульном листе пояснительной записки проекта.

При отрицательных выводах рабочей комиссии вопрос о допуске рассматривается на заседании кафедры информатики. При

отрицательном заключении кафедры протокол заседания представляется заведующему отделением программирования, который подает его для утверждения ректору, после чего учащийся информируется о том, что он не допущен к защите.

Рецензент обязан ознакомиться с материалом дипломного проекта и дать письменную рецензию. Рецензия должна содержать заключение о соответствии дипломного проекта заданию и оценку качества выполнения отдельных разделов проекта, графической части и пояснительной записки в целом. В ней должны быть указаны положительные стороны и основные недостатки проекта.

Внесение изменений в ДП после рецензирования не допускается.

Допуск к защите объявляется приказом ректора.

К защите дипломного проекта перед ГKK учащийся представляет:

- пояснительную записку;
- графические материалы;
- отзыв руководителя;
- магнитный или оптический носитель с этикеткой (при электронном представлении);
- удостоверяющий лист;
- рецензию;
- справку о внедрении, если оно есть (диплом с практическим применением).

Электронная форма представления ДП осуществляется в соответствии с «Положением об электронном представлении учебных и методических материалов», утвержденным ректором МГВРК.

Электронный вариант ДП представляет собой:

- печатный вариант пояснительной записки с приложениями;
- удостоверяющий лист (рисунок 1);
- отзыв руководителя на стандартном бланке;
- рецензию на ДП;
- магнитный или оптический носитель в конверте с этикеткой (рисунок 2), содержащий пояснительную записку, текст программы, графическую часть проекта.

К защите дипломник подготавливает материалы для компьютерной презентации ДП в режиме слайд-шоу и использует их при защите.

Удостоверяющий лист
электронного документа – дипломный проект

Тема (ДП) Программно-педагогическое средство «Жизнь и творчество А. С. Пушкина»

Обозначение (ДП) ДП8384.017209.091

Разработчик Николаев В. Р. Зав.кафедрой Скудняков Ю. А.
(Ф.И.О.) (Ф.И.О.)

Руководитель Смирнов Н. А. Рецензент _____
(Ф.И.О.) (Ф.И.О.)

Нормоконтроль Петров П. П. Зав.отделением Орешко А. А.
(Ф.И.О.) (Ф.И.О.)

Подписи лиц, ответственных за разработку ЭД

<i>Состав ЭД</i>	<i>Разработчик</i>	<i>Руководитель</i>
Пояснительная записка (на бумажном носителе формата А4), файл ПЗ.doc	подпись	подпись
Чертеж №1, файл Z1	подпись	подпись
Чертеж №2, файл Z2	подпись	подпись
Программная документация: файл текст программы		
Тип носителя: магнитный		

Секретарь ГKK _____
(подпись / дата)

Рисунок 1 – Пример заполнения удостоверяющего листа

Этикетка для дипломных (курсовых) проектов
Дипломный проект
(название) <u>Программно-педагогическое средство «Жизнь и творчество А. С. Пушкина»</u>
ДП8384.017209.091
Разработан <u> </u> дата <u> </u>
Утвержден <u> </u> дата допуска к защите <u> </u>
Разработчик: <u>Николаев В. Р.</u>
Руководитель: <u>Смирнов Н. А.</u>
Технические средства: <u>минимальный состав аппаратных средств</u>
Программные средства: <u>специальные программные средства для работы программы</u>
Состав документа: пояснительная записка – файл (Имя файла) ПЗ.doc программные документы – файлы «Имя файла 1», «Имя файла 2», ... графическая часть – файлы («Имя файла 1»), («Имя файла 2»), Z1, Z2
Сведения о защите информации: пароль 8384

Рисунок 2 – Пример заполнения этикетки

Процедура защиты дипломного проекта устанавливается председателем ГKK и включает, как правило:

- доклад учащегося (не более 10 минут);
- чтение отзыва и рецензии;
- вопросы членов ГKK, которые могут касаться темы проекта или носить общий характер в пределах дисциплин специальности;
- ответы дипломника.

Учащемуся, защитившему дипломный проект, присваивается квалификация техника-программиста.

Дипломный проект после защиты хранится в колледже 5 лет.

3 ОБЩИЕ ТРЕБОВАНИЯ К ДИПЛОМНОМУ ПРОЕКТИРОВАНИЮ

Дипломный проект состоит из пояснительной записки (ПЗ) и графической части. Объем ПЗ должен быть не менее 40 страниц (без учета приложений), объем графической части – не менее 2-х листов формата А1. Возможен электронный вариант.

Состав пояснительной записки:

- титульный лист установленного образца;
- задание;
- содержание;
- введение;
- разделы технической части;
- раздел экономической части;
- раздел охраны труда;
- заключение;
- литература;
- приложения.

Введение должно быть кратким и четким. В нем не должно быть общих фраз и отступлений, не связанных с разрабатываемой темой. Во введении обосновывается актуальность темы, дается краткий анализ достижений в области, посвященной теме ДП, формулируются цель дипломного проекта и задачи, решение которых необходимо для достижения этой цели.

Заключение содержит перечисление основных результатов, характеризующих полноту решения поставленных задач и подводящих итог содержания ДП. Результаты следует излагать в форме констатации фактов с использованием слов «изучены», «сформулированы», «разработаны», «показаны», «предложены», «подготовлены» и т. п. Текст должен быть кратким и ясным.

Пояснительная записка оформляется в соответствии с требованиями, изложенными в [11].

3.1 Требования к технической части дипломного проекта

Содержание технической части зависит от вида дипломного проекта и должно иметь объем не менее 20 страниц. Дипломные проекты условно делятся на следующие виды:

1) программные компоненты системного характера: компиляторы, интерпретаторы, системные программы, выполняющие различные вспомогательные функции, драйверы, программы-оболочки, утилиты (под утилитами имеются в виду программы резервирования, программы копирования информации, программы-упаковщики (архиваторы), программы-русификаторы, программы для диагностики компьютеров, программы-КЭШи (для ускорения доступа к информации на дисках), программы для оптимизации дисков, программы динамического сжатия дисков, программы ограничения доступа (для защиты информации));

2) программные компоненты прикладного характера: автоматизированные системы, программные средства учебного назначения, компьютерные игры, реализация математических моделей, расчетные и информационные задачи, сетевые программы, интернет-приложения.

Содержание разделов дипломного проекта зависит от выбранного подхода к реализации программного средства. Различают структурный, объектный и смешанный подходы.

Сущность структурного подхода к разработке программного обеспечения (ПО) заключается в его декомпозиции (разбиении) на автоматизируемые функции: система разбивается на функциональные подсистемы, которые, в свою очередь, делятся на подфункции, те – на задачи и так далее до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимосвязаны. Структурный подход базируется на двух основополагающих принципах:

- 1) использование процедурного стиля программирования;
- 2) последовательная декомпозиция алгоритма решения задачи сверху вниз.

Объектно-ориентированное программирование (ООП) – это методика разработки программ, в основе которой лежит понятие «объект». Объект – это некоторая структура, соответствующая объекту реального мира, его поведению. Задача, решаемая с использованием методики ООП, описывается в терминах объектов и операций над ними, а программа при таком подходе представляет собой набор объектов и связей между ними.

Смешанное программирование в настоящее время является наиболее распространенным способом написания программных компонент и объединяет различные подходы к проектированию

и разработке программ. Одной из разновидностей смешанного подхода является визуальное программирование. Диалекты различных языков используют визуальные методы для построения интерфейсов, но элементы управления могут быть разработаны как структурным способом, так и объектным.

3.2 Рекомендуемая структура технической части дипломного проекта

Руководитель ДП по своему усмотрению может добавлять разделы технической части ПЗ, структурировать их на подразделы, пункты, подпункты.

В общем случае техническая часть ПЗ к ДП должна содержать описание следующих этапов создания программного средства:

- анализ предметной области и формулировку требований к программе;
- проектирование задачи;
- реализацию программы в кодах (собственно программирование);
- тестирование программы;
- руководство по применению программы.

Первый раздел содержит анализ и постановку задачи: исследование предметной области, определение круга задач, который должна решать программа, выделение ключевых абстракций, обоснование необходимости компьютерной обработки информации или процессов, анализ существующих решений.

Второй раздел отражает этап проектирования задачи, который заключается в разработке моделей, позволяющих понять структуру будущей системы и процесс ее функционирования, сбалансировать требования и наметить схему ее применения, определить виды выходных данных, их представление и способы хранения, вычислительную систему, алгоритм решения задачи (при структурном подходе); описание классов, иерархию классов, механизмы взаимодействия классов (при объектном подходе); функциональная модель объекта, модель данных, организация данных – логическая и физическая структура данных в контексте инструмента разработки; процессы; описание пользовательского интерфейса.

Результатом должна стать детализированная модель системы, представленная в виде совокупности диаграмм в нотации

UML. Согласно методологии объектно-ориентированного анализа и проектирования, именно данная модель должна «служить» исходной информацией для написания программного кода. Итогом работы должна стать некоторая концептуальная схема, содержащая описание основных компонентов и тех функций, которые они должны выполнять.

Третий раздел (программирование) содержит описание программы. Здесь освещаются:

- обоснование выбранных средств и инструментов разработки;
- требования к аппаратным ресурсам и ОС;
- решение конкретной задачи проекта выбранными средствами.

Этап программирования вряд ли нуждается в уточнении, поскольку является наиболее традиционным для программистов. Появление инструментариев быстрой разработки приложений (Rapid Application Development (RAD)) позволило существенно сократить время и затраты на выполнение этого этапа. Результатом данного этапа является программное приложение, которое обладает требуемой функциональностью и способно решать нужные задачи в конкретной предметной области.

Четвертый раздел определяет содержание испытаний программного средства. На этапе тестирования программы проводится проверка работоспособности программы на некоторой совокупности исходных данных или при некоторых специальных режимах эксплуатации. Результатом является повышение надежности программы, исключаящее возникновение критических ситуаций или нанесение ущерба компании, использующей данное приложение.

Пятый раздел содержит сведения о применении программы. Здесь описываются:

- назначение и область применения;
- используемые технические средства и ОС;
- вызов и загрузка;
- входные данные;
- выходные данные;
- установка программы;
- диалог с пользователем;
- система помощи;
- наличие демонстрационного ролика;
- руководство пользователя.

3.2.1 Содержание разделов технической части дипломного проекта при разработке автоматизированных систем

Пример формулировки исходных данных при проектировании автоматизированного рабочего места (АРМ) приведен на рисунке 3.

Міністэрства адукацыі Рэспублікі Беларусь УА «Мінскі дзяржаўны вышэйшы радыётэхнічны каледж»
АДДЗЯЛЕННЕ <u>праграмавання</u> Спецыяльнасць _____ гр. _____
ЗАЦВЯРДЖАЮ Загадчык кафедры _____ (Ю.А.Скудняков) « ____ » _____ 2007г.
ЗАДАНИЕ на дыпломнае праектаванне
Дыпломніку _____ 1. Тэма праекта <u>Автоматизированное рабочее место заведующего отделением</u> Замацавана загадам па каледжу ад _____ № _____ 2. Тэрмін здачы дыпломнікам закончанага праекта _____ 3. Зыходныя данныя па праекту <u>списки групп учащихся отделения по специальностям, содержащие по каждому студенту: общие признаки (Ф.И.О., год, месяц, день рождения, группа), дополнительные сведения (место жительства, телефон)), сведения об успеваемости по семестрам.</u> 1. Тлумачальная запіска <u>Введение 1 Назначение и цели создания системы 2 Характеристика объекта автоматизации 2.1 Сведения об объекте автоматизации 2.2 Обоснование необходимости компьютерной обработки информации 3 Логическая модель данных 4 Физическая модель 4.1 Обоснование средств разработки 4.2 Описание доменов 4.3 Описание таблиц 4.4 Описание ограничений 4.5 Описание индексов 4.6 Описание запросов к БД 5 Описание программы 5.1 Обоснование среды разработки 5.2 Описание интерфейса 5.3 Программно-аппаратные ресурсы ПК 6 Тестирование 7 Применение 7.1 Назначение программы 7.2 Требования к аппаратным ресурсам ПК 7.3 Руководство пользователя 8 Экономический раздел 9 Охрана труда Заключение Приложения</u> 2. Графічная частка праекта Ліст 1 Диаграмма деятельности _____ Ліст 2 Схема работы системы _____

Рисунок 3 – Пример исходных данных при разработке АРМ заведующего отделением

Раздел «Назначение и цели создания системы» может состоять из следующих подразделов:

- назначение системы. Здесь указываются: вид автоматизированной деятельности и перечень объектов автоматизации, на которых предлагается использовать разрабатываемую систему;
- цели создания системы;
- определение круга задач, которые должны быть автоматизированы.

Раздел «Характеристика объекта автоматизации» должен содержать:

- краткие сведения об объекте автоматизации;
- сведения об условиях эксплуатации объекта автоматизации;
- обоснование необходимости компьютерной обработки информации.

Раздел «Логическая модель данных» содержит: полное развернутое описание предметной области, модели предметной области, модель «сущность-связь» и ее описание, описание отношений, разрабатываемых на базе диаграммы «сущность-связь».

Раздел «Физическая модель». Здесь отношения, разработанные на стадии формирования логической модели данных, преобразуются в таблицы. В этом подразделе необходимо описать:

- выбор и обоснование средств разработки;
- описание доменов;
- описание таблиц;
- описание ограничений;
- выбранные индексы;
- описание запросов к БД.

Раздел «Описание программы». Здесь должны быть отражены:

- выбор и обоснование среды разработки;
- описание интерфейса;
- программно-аппаратные ресурсы ПК.

Раздел «Тестирование». В разделе необходимо привести описание способов проверки, позволяющих дать общее заключение о работоспособности программы (контрольные примеры, результаты).

Раздел «Применение». Здесь необходимо указать назначение программы, условия выполнения программы (минимальные и рекомендуемые требования к аппаратно-программному обеспечению), последовательность действий пользователя, обеспечивающих загрузку, запуск, выполнение и завершение программы.

Приложения. В обязательных приложениях рекомендуется приводить: текст программы, диаграммы (вариантов использования, деятельности, последовательности и «сущность-связь»).

3.2.2 Содержание разделов технической части дипломного проекта при разработке программных средств учебного назначения

Пример формулировки исходных данных при проектировании компьютерного средства обучения приведен на рисунке 4.

Міністэрства адукацыі Рэспублікі Беларусь УА «Мінскі дзяржаўны вышэйшы радыётэхнічны каледж»
Аддзяленне <u>праграмавання</u> Спецыяльнасць _____ гр. _____
ЗАЦВЯРДЖАЮ Загладчык кафедры _____ (Ю. А. Скудняков) « ____ » _____ 2007 г.
ЗАДААННЕ на дыпломнае праектаванне
Дыпломніку _____
1. Тэма праекта <u>Компьютерное средство обучения по творчеству А. С. Пушкина</u> Замацавана загадам па каледжу ад _____ № _____
2. Тэрмін здачы дыпломнікам закончанага праекта _____
3. Зыходныя данныя па праекту <u>биография А. С. Пушкина, стихи, проза, иллюстрации, портреты, сценарии КСО, вопросы тестов.</u>
1. Тлумачальная запіска
<u>Введение 1 Назначение и область применения обучающе-тестирующей программы 2 Сценарий обучения 3 Проектирование задачи 3.1 Обоснование инструментов разработки 3.2 Описание алгоритма решения задачи 4 Программа решения задачи 4.1 Логическая структура 4.2 Физическая структура 5 Тестирование 6 Применение 6.1 Назначение программы 6.2 Требования к аппаратным ресурсам ПК 6.3 Руководство пользователя 7 Экономический раздел 8 Охрана труда Заключение Приложения</u>
2. Графічная частка праекта
Ліст 1 Схема работы системы _____
Ліст 2 Диаграмма вариантов использования _____

Рисунок 4 – Пример исходных данных при разработке программных средств учебного назначения

К программным средствам учебного назначения относятся:

- обучающе-тестирующие программы;
- игровые программы;
- программы для внеклассной работы.

Для выполнения дипломного проекта по разработке обучающе-тестирующей программы необходимо иметь глубокие знания предмета изложения, методики подачи материала и обеспечить максимум наглядности. При этом необходимо учитывать, что нецелесообразно приводить в обучающе-тестирующей программе материал, содержащий много текста.

Раздел «Назначение и область применения обучающе-тестирующей программы». Здесь необходимо указать: наименование дисциплины (раздела и темы), специальность, данные об области применения программы, аудиторию, на которую она рассчитана, общую характеристику вычислительных средств, для которых разрабатывается программа.

Раздел «Сценарий обучения». Сюда включены: содержание обучения и алгоритм управления познавательной деятельностью обучаемых, формат экрана, способ и методика подачи текста, рисунков, анимаций и звука. Пример формата экрана приведен на рисунке 5, сценария программного педагогического средства – в таблицах 1 и 2. Для контролирующего компонента определяется матрица вопросов, степень их неповторяемости, критерии выставления оценок. При этом необходимо обратить внимание на объективность выставляемой оценки.

2 – постоянная информация (например, наименование компьютерного средства обучения (КСО))	
1 – графическая зона	3 – текстовая зона
4 – зона меню	

Рисунок 5 – Пример формата экрана

Таблица 1 – Сценарий по охране труда в цехе

Номер кадра	Видео	Текст (звук)	Время, с
1	Крупным планом вывеска «Цех обмена и обработки почты» Общий план цеха: 1) транспортеры для люковых окон универсальных с выдвижными секциями (ТЛУ 2.2, ТВС-3); 2) стационарные транспортеры (ТПК, ТЛС-400); 3) установки для сортировки посылок УСП-К; 4) платформа с почтовыми вагонами; 5) работники цеха перевозят посылки, мешки с почтой на тележках; 6) общий порядок работы людей	К самостоятельной работе в качестве: - оператора; - сортировщика почтовых отправок и периодической печати; - оператора по обмену почты с почтовыми вагонами; - а также к работе на универсальных транспортерах для люковых окон с выдвижными секциями (ТЛУ 2.2, ТВС-3); - стационарных транспортерах (ТПК, ТЛС-400); - и установке для сортировки посылок УПС-К допускаются лица не моложе 18 лет, прошедшие: медицинский осмотр, инструктаж, обучение правилам техники безопасности, пожарной безопасности, проверку знаний и изучившие настоящую инструкцию, имеющие 1 группу допуска по электробезопасности и ознакомленные с соответствующими инструкциями по эксплуатации электрооборудования	30
2	Крупный план знака «Осторожно! Высокое напряжение» на электрошите, движущаяся лента конвейера, работники, перегружающие мешки	Опасными и вредными производственными факторами при выполнении работы являются: - высокое электрическое напряжение; - вращающиеся части и механизмы оборудования; - физические перегрузки (периодическое удержание груза руками); - внутрипроизводственный транспорт; - интенсивное движение поездов на станции «Минск-Пассажирский»	20
3	Актер готовится к работе в гардеробной: надевает халат, застегивает пуговицы, убирает волосы	Перед началом работы работник обязан: привести в порядок спецодежду (застегнуть пуговицы), волосы убрать под головной убор (берет или косынку и др.)	20
4	Актер подходит к УСП-К, осматривает пульт управления со всех сторон	При работе на установке УСП-К и другими транспортерами работник обязан внешним осмотром проверить: ...	25
...

Таблица 2 – Сценарий программного педагогического средства

Иллюстрация	Текст (звук)	Примечание
1 Общий вид редуктора в закрытом корпусе	Редуктором называют зубчатый, червячный, зубчато-червячный механизм, выполненный в закрытом корпусе и предназначенный для понижения угловой скорости и повышения вращательного момента	
2 Общий вид редуктора в разобранном виде	Одноступенчатый горизонтальный редуктор с цилиндрическими косозубыми колесами	
3 Общий вид корпуса редуктора	Редуктор состоит из литого чугунного корпуса	
4 Вид корпуса с подшипниковыми узлами. Подшипниковые узлы выделены зеленым цветом	В корпусе смонтированы подшипниковые узлы, служащие опорами для валов редуктора	
5 Вид корпуса с подшипниковыми узлами с маслосодержательными шайбами и прокладками	Узлы снабжены маслосодержательными шайбами и прокладками	
...

Раздел «Проектирование задачи». Здесь приводятся: описание алгоритма решения задачи, реализация сценария в терминах структурного или объектного подхода к проектированию программного средства, выбор и обоснование инструментов разработки.

Раздел «Программа решения задачи». Здесь описывается реализация сценария в терминах и понятиях выбранного средства разработки, включая сведения о логической (структура данных, функции и их действия, классы и их иерархия), физической (модули интерфейса и модули реализации) структуре и функционировании программы.

Раздел «Тестирование». Здесь освещаются подготовка определенных тестов и моделирование действий пользователя, при этом необходимо учитывать возможные неправильные действия пользователя.

Раздел «Применение». Здесь содержатся инструкции по работе с программой, описание действий пользователя при применении программы на уроке, изложении нового материала и тестировании.

В приложениях приводятся: текст программы и бумажные варианты электронных чертежей.

3.2.3 Содержание разделов технической части при разработке программных средств создания Internet-приложений

Пример формулировки исходных данных при проектировании Web-приложения приведен на рисунке 6.

<p>Міністэрства адукацыі Рэспублікі Беларусь УА «Мінскі дзяржаўны вышэйшы радыётэхнічны каледж»</p>	
<p>Аддзяленне <u>праграмавання</u> Спецыяльнасць _____ гр. _____</p>	
<p>ЗАЦВЯРДЖАЮ Загадчык кафедры _____ (Ю. А. Скудняков) «___» _____ 2007 г.</p>	
<p>ЗАДААННЕ на дыпломнае праектаванне</p>	
<p>Дыпломніку _____</p>	
<p>1. Тэма праекта <u>Интернет-магазин косметической продукции</u> Замацавана загадам па калледжу ад _____ № _____</p>	
<p>2. Тэрмін здачы дыпломнікам закончанага праекта _____</p>	
<p>3. Зыходныя данныя па праекту <u>Рубрикация, номенклатура товаров (духи, пудра, помада), характеристики товаров, сведения о поставках, сведения о производителях, цены на товары</u></p>	
<p>1. Тлумачальная запіска <u>Введение 1 Постановка задачи 1.1 Назначение системы 1.2 Цели создания Web-приложения 1.3 Структура сайта 2 Обоснование инструментов разработки 3 Реализация 3.1 Логическая структура Web-приложения 3.2 Физическая структура Web-приложения 3.3 Описание структуры и формата страниц 3.4 Описание ограничений доступа к данным 3.5. Описание используемых библиотек и элементов управления 3.6 Описание используемых функций и процедур 4 Тестирование 5 Применение 5.1 Назначение и область применения программы 5.2 Описание организации диалога с пользователем 5.3 Программно-аппаратное обеспечение сервера и клиента 5.4 Руководство пользователя 6 Экономический раздел 7 Охрана труда Заключение Приложения</u></p>	
<p>2. Графічная частка праекта</p>	
<p>Ліст 1 Дыяграма паслядоўнасці _____</p>	
<p>Ліст 2 Дыяграма кооперацыі _____</p>	

Рисунок 6 – Пример исходных данных при разработке программных средств создания Internet-приложений

Раздел «Постановка задачи». Здесь указываются: цель создания Web-приложения, задачи, которые планируется решать посредством сайта; определяется потенциальная аудитория; описываются структура сайта и требования к разрабатываемому приложению.

Раздел «Объектно-ориентированное проектирование» включается в состав технической части ПЗ, если для разработки Web-приложения используется объектно-ориентированный подход. Здесь содержится описание классов, их иерархии, а также механизмов взаимодействия классов.

Раздел «Выбор инструментов разработки». Здесь необходимо обосновать выбранный инструмент разработки сайта.

В разделе «Реализация» описываются:

- логическая структура Web-приложения (структура страниц, связи между ними), графическое представление логической структуры (карта сайта), где прямоугольниками обозначены страницы, а линиями – связи между ними, содержание и структура главной страницы;

- физическая структура Web-приложения с информацией о размещении файлов сайта, перечень каталогов;

- структура и формат страниц, деление текста на строки и абзацы, цветовая разметка страницы, фоновое изображение, звук, наличие и содержание бегущей строки, наличие фреймов, таблиц, гиперссылок;

- используемые библиотеки и элементы управления;

- используемые процедуры и функции;

- ограничения доступа к данным.

В разделе «Тестирование» необходимо отразить проверку:

- правильности работы гиперссылок, времени загрузки страниц, совместимости сайта с различными браузерами и разными настройками браузеров (при отключении выполнения скриптов, элементов управления ActiveX);

- правильности взаимодействия Web-приложения с пользователем (необходимо указать конкретные тестовые ситуации и результаты, к которым они привели).

Раздел «Применение». Здесь указываются назначение сайта, область применения, организация диалога с пользователем, способы ввода информации, возможности управления работой сайта, программно-аппаратное обеспечение сервера и клиента для корректной работы Web-приложения.

В приложениях обязательно помещать текст программы, бумажные варианты электронных чертежей.

3.3 Графическая часть ДП

Графическая часть дипломного проекта в зависимости от темы выполняется, как правило, на листах чертежной бумаги формата А1 в соответствии с требованиями стандартов. При электронном представлении проекта бумажные копии электронных чертежей помещают в приложениях. Схемы выполняются в соответствии с требованиями [14].

Для дипломных проектов рекомендуются следующие виды чертежей, схем, диаграмм: схемы алгоритмов, программ, данных и систем (далее схемы), различные виды диаграмм.

Схемы данных отображают путь данных при решении задач и определяют этапы обработки, а также различные применяемые носители данных. Схема данных состоит из символов данных (символы данных могут также указывать вид носителя данных); символов процесса, который следует выполнить над данными (символы процесса могут также указывать функции, выполняемые ЭВМ); символов линий (линейных символов) и специальных символов, используемых для облегчения написания и чтения схемы. Символы данных предшествуют и следуют за символами процесса. Схема данных начинается и заканчивается символами данных (за исключением специальных символов – «соединитель» и «терминатор»). Пример приведен на рисунке 7.

Схемы программ отображают последовательность операций в программах. Схема программы состоит из символов процесса, указывающих фактические операции обработки данных; линейных символов, указывающих поток управления, и специальных символов, используемых для облегчения написания и чтения схемы программы. Пример приведен на рисунке 8.

Схемы работы системы отображают управление операциями и поток данных в системе. Схема работы системы состоит из символов данных, указывающих на наличие данных; символов процесса, указывающих операции над данными; линейных символов, указывающих потоки данных и поток управления, и специальных символов, используемых для облегчения написания и чтения блок-схемы. Пример приведен на рисунке 9.

Схемы взаимодействия программ отображают путь активаций программ и взаимодействий с соответствующими данными. Каждая программа в схеме взаимодействия программ показывается только один раз (в схеме работы системы программа может изображаться более чем в одном потоке управления). Схема взаимодействия про-

грамм состоит из символов данных, символов процесса, линейных символов и специальных символов. Пример приведен на рисунке 10.

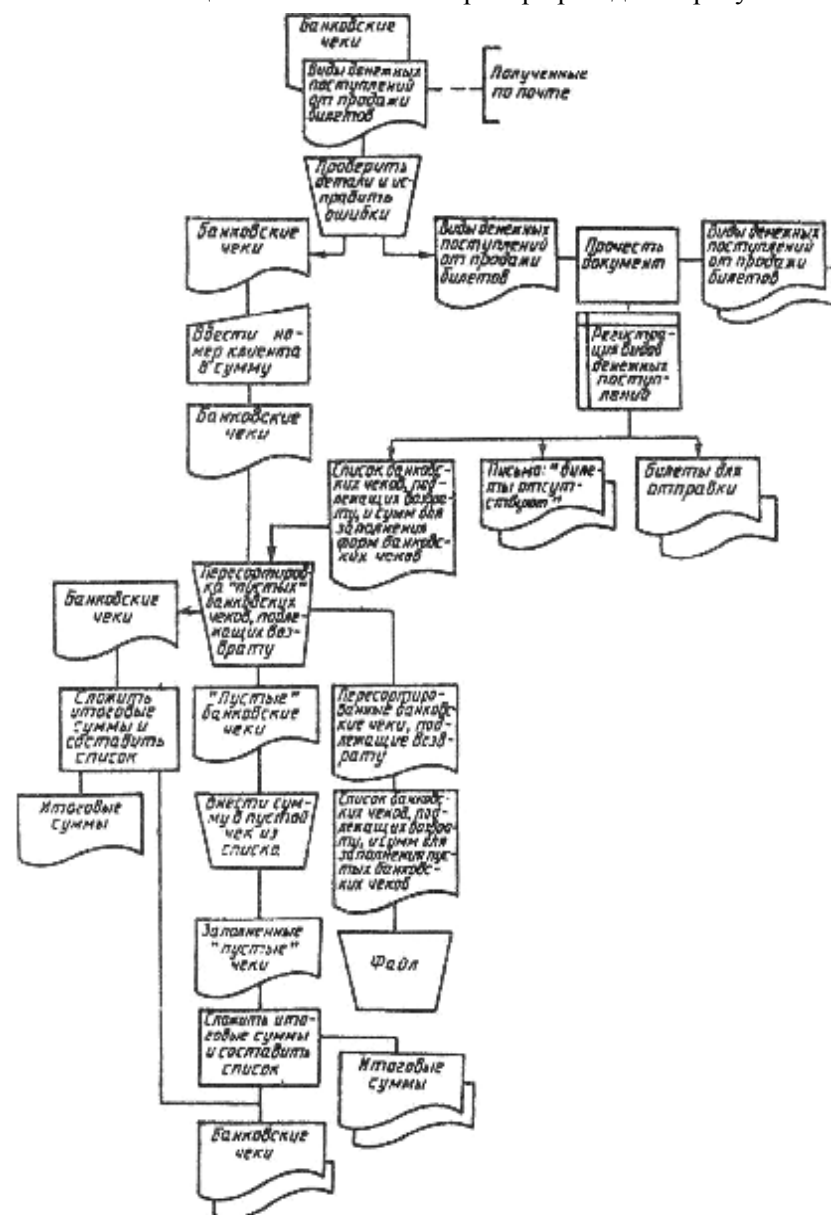


Рисунок 7 – Схема данных

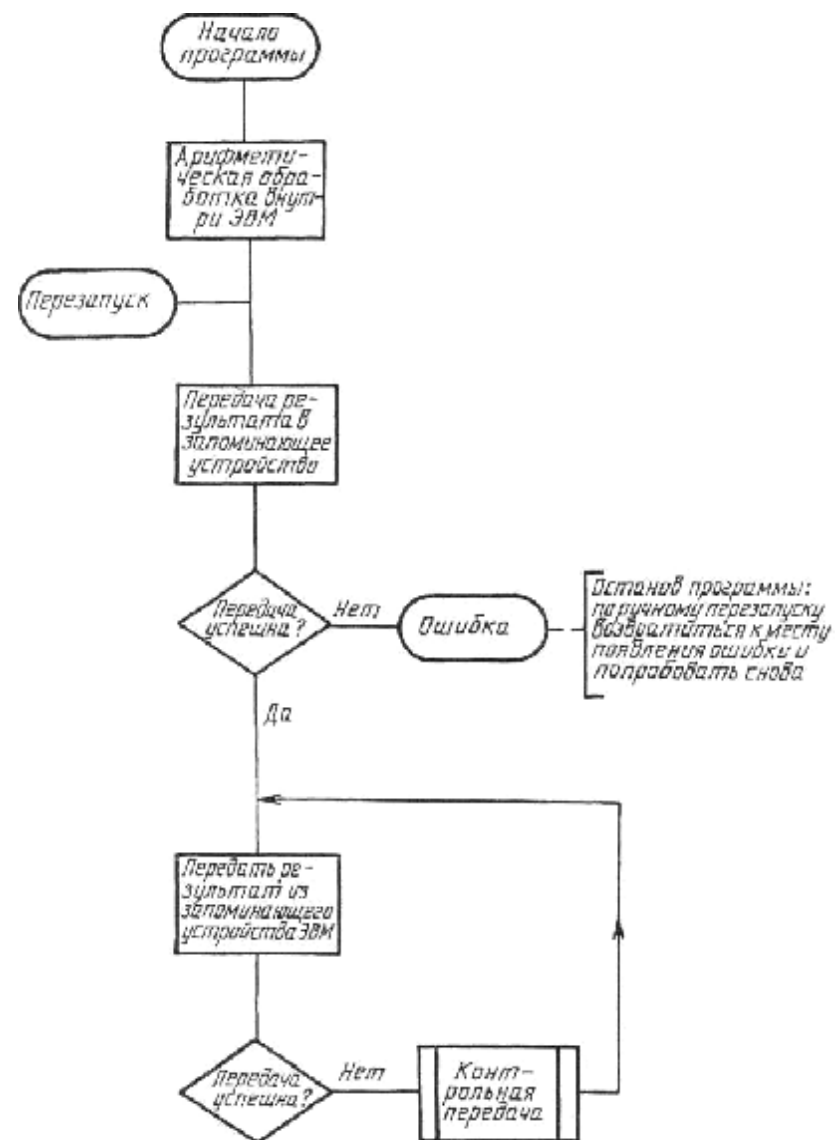


Рисунок 8 – Схема программ

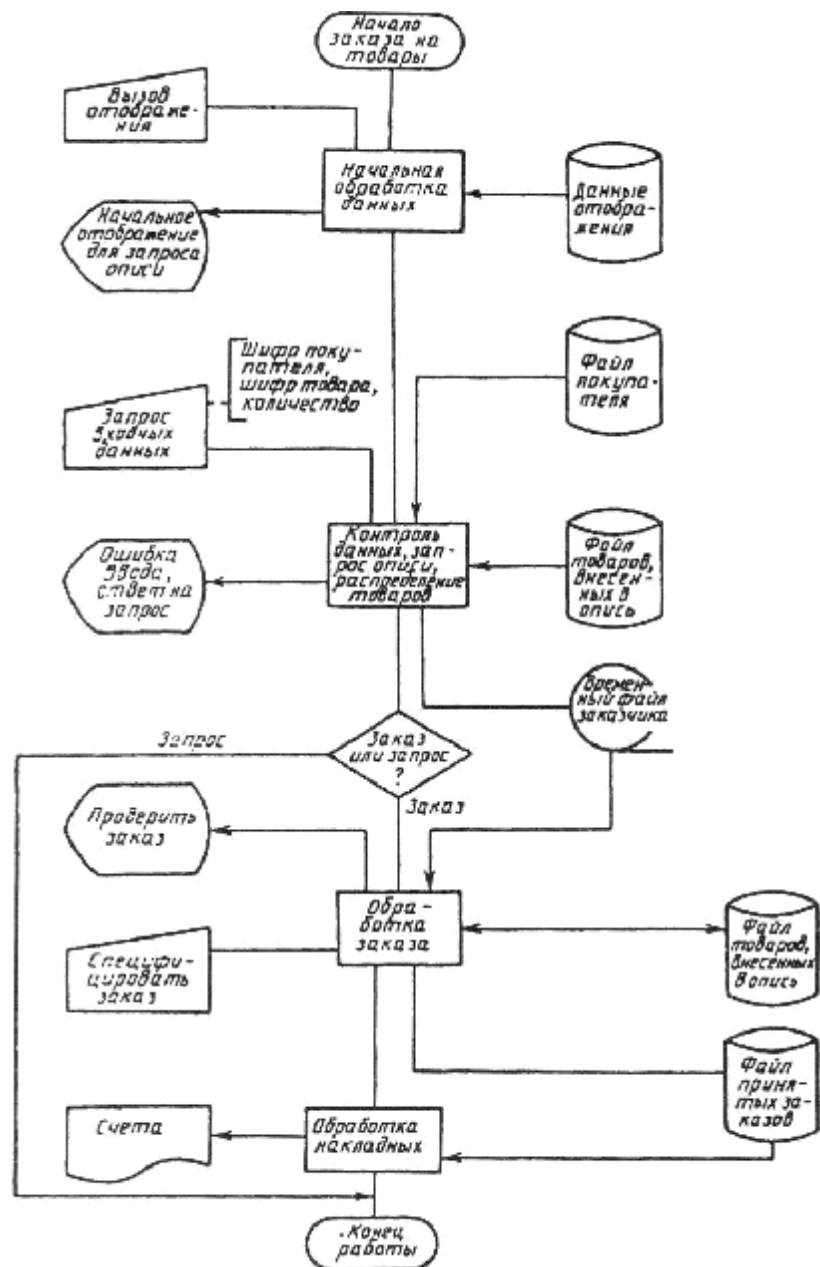


Рисунок 9 – Схема работы системы

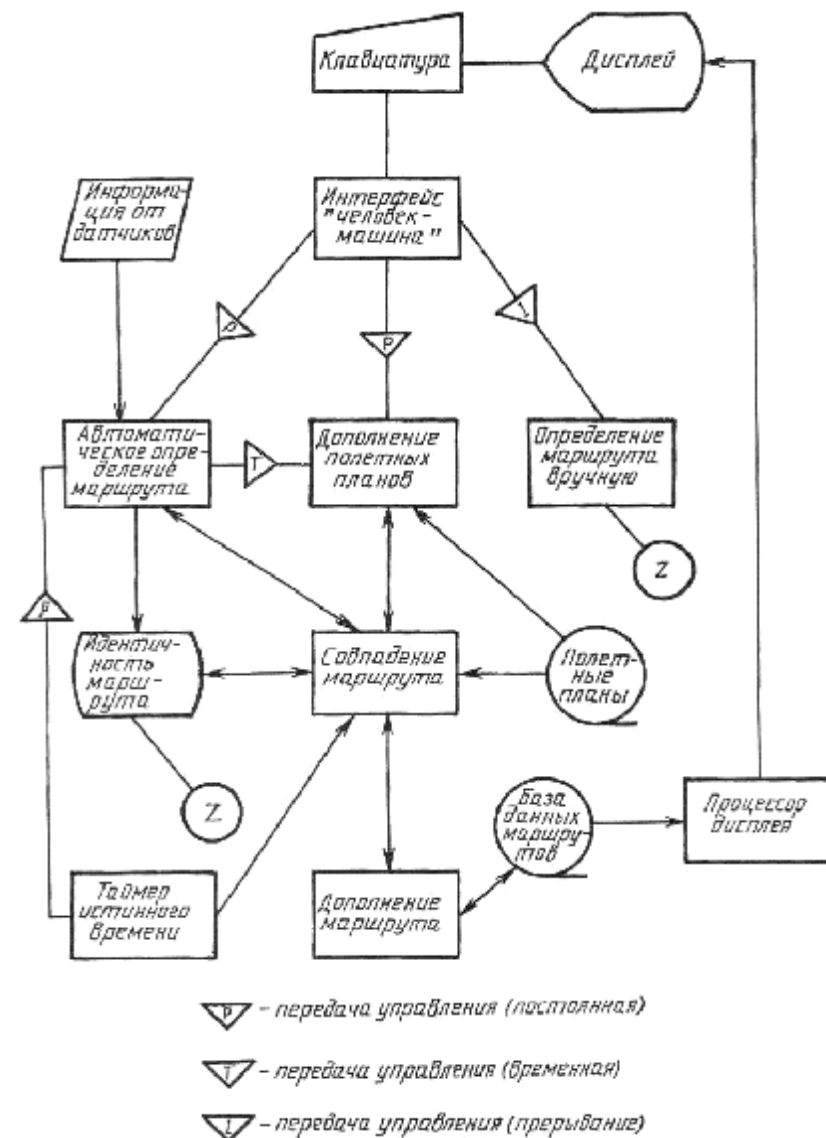


Рисунок 10 – Схема взаимодействия программ

```

graph TD
    FT((Файл транзакций)) --- K[Kорректура]
    TDT(Таблица проверки достоверности) --- K
    KF[(Рабочий файл)] <--> K
    GF[(Главный файл)] <--> K
    K --- O[Отчет]
    K --- SO[Список ошибок]
  
```

Для графической части рекомендуются следующие виды диаграмм:

- диаграмма вариантов использования;
- диаграмма классов;
- диаграмма состояний;
- диаграмма деятельности;
- диаграмма последовательности;
- диаграмма кооперации;
- диаграмма компонентов.

Диаграммы вариантов использования описывают функциональное назначение системы или то, что система должна делать. Пример приведен на рисунке 12.

```

    usecaseDiagram
        actor Operator
        actor Employer as Наниматель
        actor Applicant as Претендент

        usecase UC1((Набор специалистов и студентов))
        usecase UC2((Подача заявки))
        usecase UC3((Поиск по базе данных))
        usecase UC4((Обеспечение информацией))
        usecase UC5((Выдача данных по запросу))
        usecase UC6((Получение данных))
        usecase UC7((Оформление заявки))
        usecase UC8((Внесение данных))

        Operator -- UC1
        Operator -- UC2
        Employer -- UC1
        Applicant -- UC2

        UC1 -.-> UC3 : <<include>>
        UC1 -.-> UC4 : <<include>>
        UC1 -.-> UC5 : <<extended>>
        UC2 -.-> UC3 : <<include>>
        UC2 -.-> UC6 : <<include>>
        UC2 -.-> UC7 : <<extended>>
        UC2 -.-> UC8 : <<extended>>

        UC1 * UC3
        UC1 * UC4
        UC2 * UC3
        UC2 * UC6
        UC2 * UC7
        UC2 * UC8
    
```

Актер представляет собой любую внешнюю по отношению к моделируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей. Обозначением актера на диаграммах является фигурка человечка, под которой записывается имя актера.

На диаграмме вариантов использования интерфейс изображается в виде маленького круга, рядом с которым записывается его имя. В качестве имени может быть существительное или строка текста. Если имя записывается на английском языке, то оно должно начинаться с заглавной буквы I.

Графический символ отдельного интерфейса соединяется на диаграмме сплошной линией или пунктирной линией со стрелкой с тем вариантом использования, который его поддерживает. Сплошная линия указывает, что связанный с интерфейсом вариант использования должен реализовывать все необходимые для него сервисы. Пунктирная линия со стрелкой означает, что вариант использования предназначен для спецификации только того сервиса, который необходим для реализации данного интерфейса.

Между элементами диаграммы вариантов использования могут существовать различные отношения, которые описывают взаимодействие экземпляров актеров и вариантов использования:

- ассоциации (association relationship);
- расширения (extend relationship);
- обобщения (generalization relationship);
- включения (include relationship).

3.3.2 Диаграмма классов

Диаграмма классов (class diagram) служит для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Пример приведен на рисунке 13.

Диаграмма классов представляет собой граф, вершинами которого являются элементы типа «классификатор», связанные различными типами структурных отношений, и содержит интерфейсы, пакеты, отношения и даже отдельные экземпляры, такие как объекты и связи.

Для изображения интерфейсов используется графический символ – прямоугольник класса с ключевым словом или стереотипом «interface». При этом секция атрибутов у прямоугольника отсутствует, а указывается только секция операций.

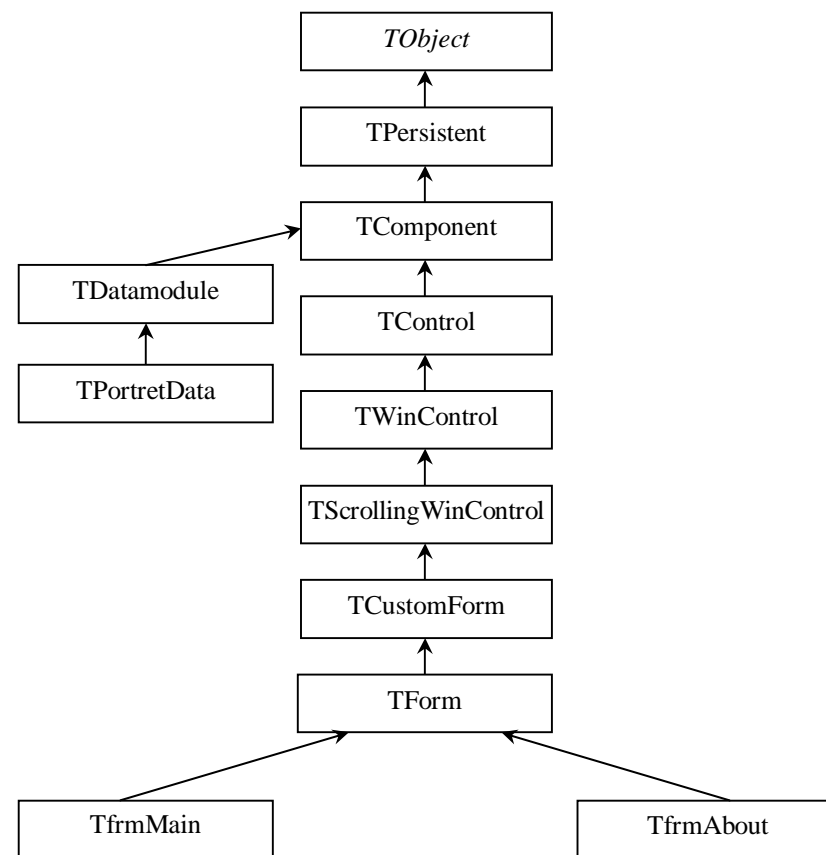


Рисунок 13 – Пример диаграммы классов

Для графического изображения объектов используется такой же символ прямоугольника, что и для классов. Имена объектов обязательно подчеркиваются.

3.3.3 Диаграмма состояний

Диаграммы состояний используются для описания поведения отдельных объектов и могут быть применены для спецификации функциональности других компонентов моделей, таких как варианты использования, актеры, подсистемы, операции и методы. Пример приведен на рисунке 14.

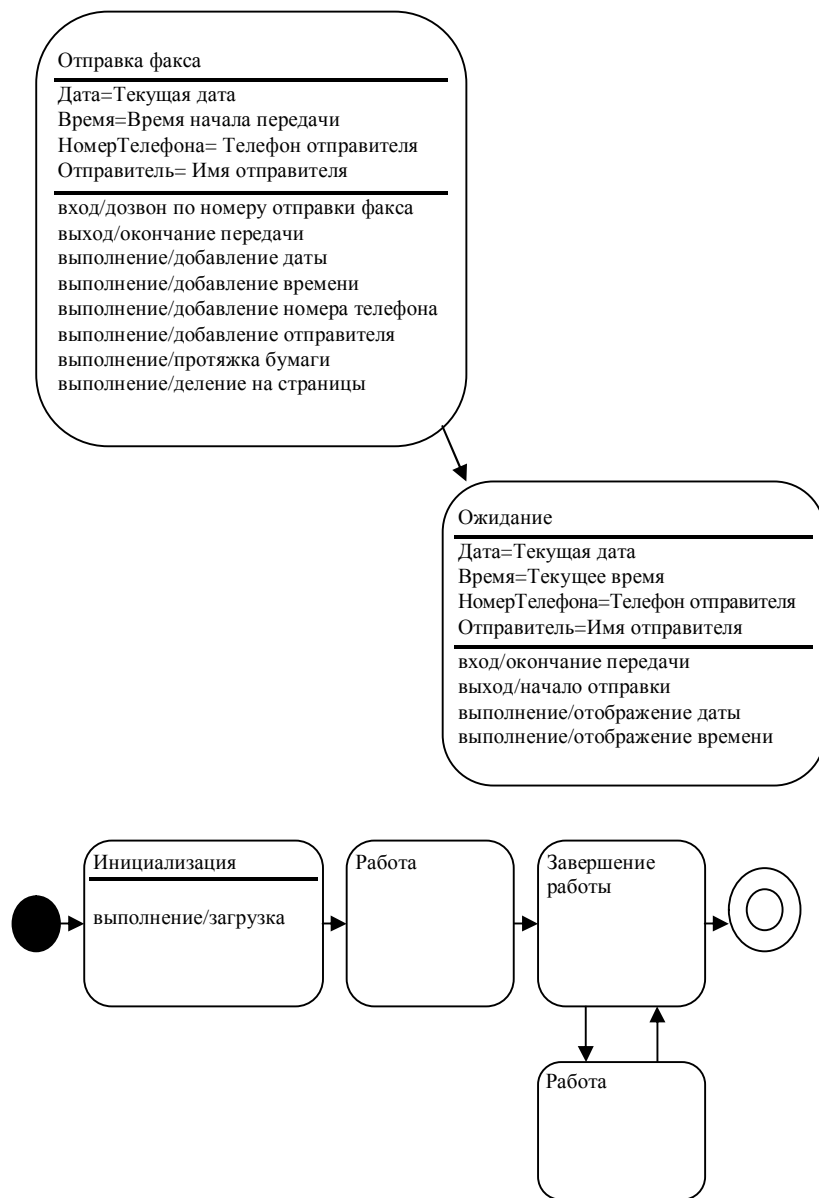


Рисунок 14 – Пример диаграммы состояний

Состояние на диаграмме изображается прямоугольником со скругленными вершинами. Прямоугольник может быть разделен на две секции горизонтальной линией.

Переход представляет собой отношение между двумя последовательными состояниями, которое указывает на факт смены одного состояния объекта другим (рисунок 15).



Рисунок 15 – Изображение составного состояния

3.3.4 Диаграмма деятельности

Для моделирования процесса выполнения операций в языке UML используются диаграммы деятельности. Диаграмму деятельности можно считать частным случаем диаграмм состояний.

Состояние действия (action state) является специальным случаем состояния с некоторым входным действием и, по крайней мере, одним выходящим из состояния переходом.

Графически состояние действия изображается прямоугольником с закругленными углами. Внутри этого изображения записывается выражение действия (action-expression), которое должно быть уникальным в пределах одной диаграммы деятельности.

Переход переводит деятельность в последующее состояние сразу, как только закончится действие в предыдущем состоянии. На диаграмме такой переход изображается сплошной линией со стрелкой.

Пример диаграммы деятельности приведен на рисунке 16.

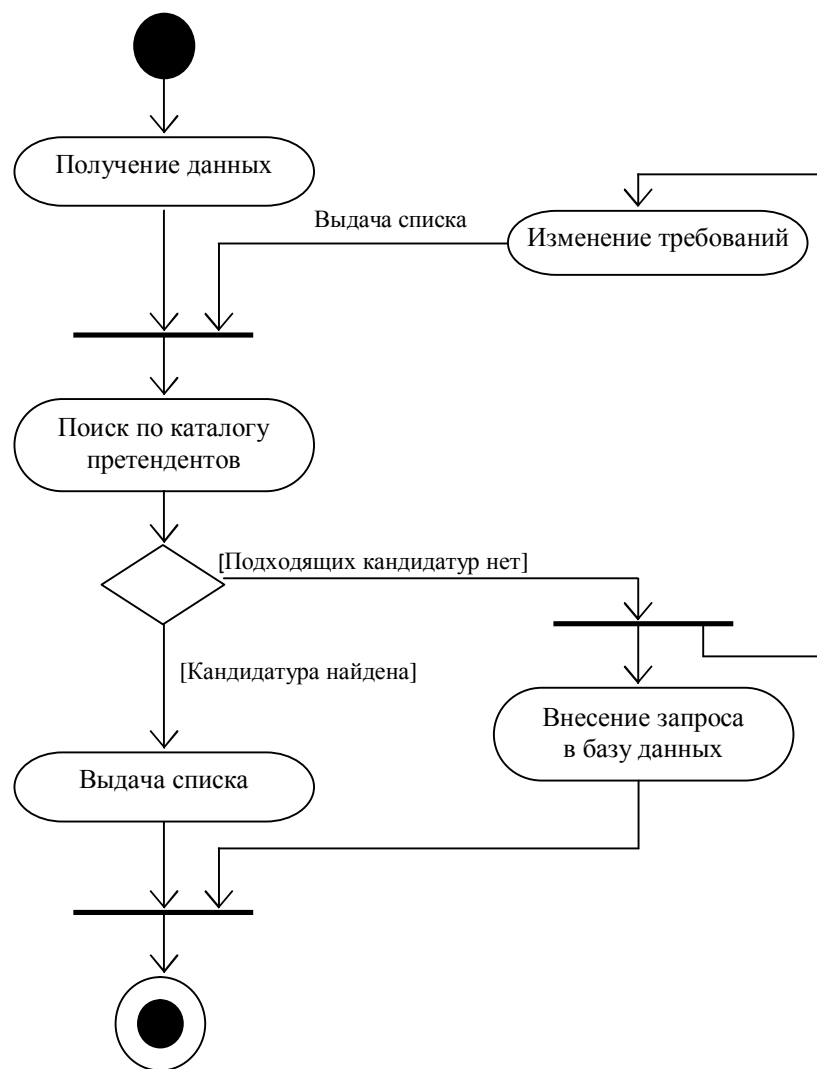


Рисунок 16 – Пример диаграммы деятельности

3.3.5 Диаграмма последовательности

Для моделирования взаимодействия объектов во времени в языке UML используются диаграммы последовательности. Пример приведен на рисунке 17.

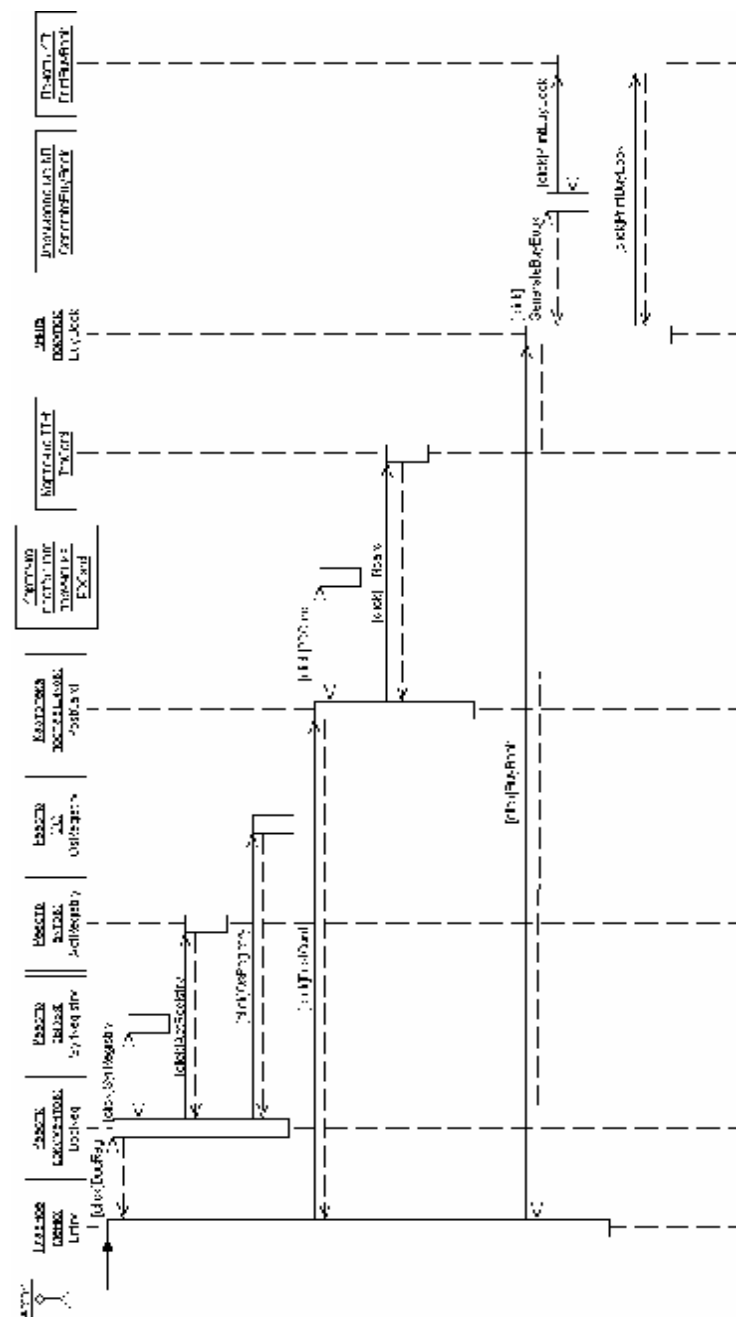


Рисунок 17 – Пример диаграммы последовательности

На диаграмме последовательности изображаются только те объекты, которые непосредственно участвуют во взаимодействии.

Графически каждый объект изображается прямоугольником и располагается в верхней части своей линии жизни. Внутри прямоугольника записываются имя объекта и имя класса, разделенные двоеточием. При этом вся запись подчеркивается, что является признаком объекта.

Вторым измерением диаграммы последовательности является вертикальная временная ось, направленная сверху вниз.

Линия жизни объекта (object lifeline) изображается пунктирной вертикальной линией.

Фокус управления изображается в форме вытянутого узкого прямоугольника, верхняя сторона которого обозначает начало получения фокуса управления объектом (начало активности), а его нижняя сторона – окончание фокуса управления (окончание активности).

Сообщения на диаграмме последовательности изображаются горизонтальными стрелками, соединяющими линии жизни или фокусы управления двух объектов.

3.3.6 Диаграмма кооперации

Диаграмма кооперации позволяет графически представить последовательность взаимодействия и все структурные отношения между объектами, участвующими в этом взаимодействии. Пример приведен на рисунке 18.

На диаграмме кооперации в виде прямоугольников изображаются участвующие во взаимодействии объекты, содержащие имя объекта, его класс и, возможно, значения атрибутов; указываются ассоциации между объектами в виде различных соединительных линий.

В отличие от диаграммы последовательности, на диаграмме кооперации изображаются только отношения между объектами, играющими определенные роли во взаимодействии.

Кооперация определяет структуру поведения системы в терминах взаимодействия участников этой кооперации.

Простой класс на диаграмме кооперации обозначается прямоугольником класса, внутри которого записывается строка текста.



Рисунок 18 – Пример диаграммы кооперации

Объект (object) является отдельным экземпляром класса, который создается на этапе выполнения программы.

Связь на диаграмме кооперации изображается отрезком прямой линии, соединяющим два прямоугольника объектов. Рядом с линией в ее средней части может записываться имя соответствующей ассоциации.

Сообщения на диаграмме кооперации специфицируют коммуникацию между двумя объектами и обозначаются одним из четырех видов стрелок:

- сплошная линия с треугольной стрелкой обозначает вызов процедуры или другого вложенного потока управления. Может быть также использована совместно с параллельно активными объектами, когда один из них передает сигнал и ожидает, пока не закончится некоторая вложенная последовательность действий.

вий. Обычно все такие сообщения являются синхронными, т. е. инициируемыми по завершении некоторой деятельности или при выполнении некоторого условия;

- сплошная линия с V-образной стрелкой обозначает простой поток управления. Каждая такая стрелка изображает один этап в последовательности потока управления. Обычно все такие сообщения являются асинхронными;

- сплошная линия с полустрелкой используется для обозначения асинхронного потока управления. Соответствующие сообщения формируются в произвольные, заранее не известные моменты времени, как правило, активными объектами. Обычно сообщения этого типа являются начальными в последовательности потока управления и чаще всего инициируются актерами;

- пунктирная линия с V-образной стрелкой обозначает возврат из вызова процедуры. Стрелки этого типа зачастую отсутствуют на диаграммах кооперации, поскольку неявно предполагается их существование после окончания процесса активизации некоторой деятельности.

3.3.7 Диаграмма компонентов

Для физического представления моделей систем используются диаграммы реализации (implementation diagrams), которые включают в себя диаграмму компонентов (рисунок 19) и диаграмму развертывания.

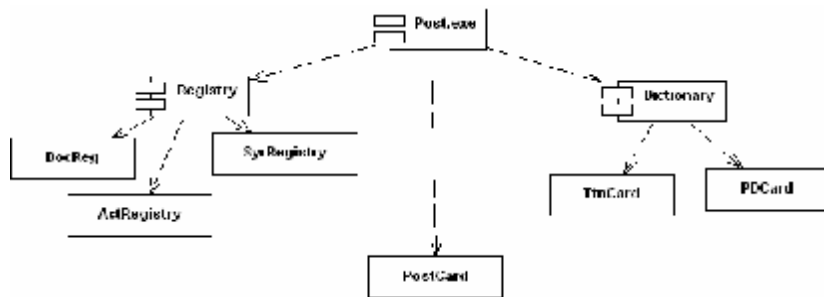


Рисунок 19 – Пример диаграммы компонентов

Диаграмма компонентов описывает особенности физического представления системы. Основными графическими элементами диаграммы компонентов являются компоненты, интерфейсы и зависимости между ними.

Графический компонент – прямоугольник со вставленными слева двумя более мелкими прямоугольниками. Внутри большого прямоугольника записывается имя компонента и, при необходимости, некоторая дополнительная информация.

Интерфейс графически изображается окружностью, которая соединяется с компонентом отрезком линии без стрелок. Имя интерфейса должно начинаться с заглавной буквы I и записываться рядом с окружностью.

Зависимость служит для представления только факта наличия такой связи, когда изменение одного элемента модели оказывает влияние или приводит к изменению другого элемента модели. Отношение зависимости на диаграмме компонентов изображается пунктирной линией со стрелкой, направленной от клиента (зависимого элемента) к источнику (независимому элементу).

Рекомендуемая литература

- 1 Бахтизин, В. В. Средства анализа и моделирования в среде Case-средств / В. В. Бахтизин, Л. А. Глухова. – Мн., 2005.
- 2 Буч, Г. Язык UML : руководство пользователя : пер. с англ. / Г. Буч, Д. Рамбо, А. Джекобсон. – М. : ДНК, 2000.
- 3 Вим, Дж. Искусство Web-дизайна : самоучитель / Дж. Вим. – СПб. : Питер, 2002.
- 4 Гофман, В. Э. Delphi 7 / В. Э. Гофман, А. Д. Хомоненко. – СПб. : Санкт-Петербург, 2003.
- 5 Гусин, А. Н. Методическое пособие по работе в Rational Rose / А. Н. Гусин, Р. Т. Хабибрахманов, В. О. Лонский. – РФ., 2005.
- 6 Дейтел, П. Как программировать для Internet & WWW / П. Дейтел, Х. Дейтел. – М. : Бином, 2002.
- 7 Иванова, Г. С. Технология программирования / Г. С. Иванова. – М. : МГТУ им. М. Э. Баумана, 2002.
- 8 Комолова, Н. В. HTML : учеб. курс / Н. В. Комолова – СПб. : Питер, 2007.
- 9 Константинов, А. В. Компьютерная графика : конспект лекций / А. В. Константинов. – Ростов н/Д : Феникс, 2006.
- 10 Орлов, С. А. Технология разработки программ / С. А. Орлов. – СПб. : Питер, 2004.
- 11 Оформление курсовых и дипломных проектов : метод. указания / сост. И. М. Снежкова. – Мн. : МГВРК, 2003.
- 12 Савитч, У. Язык C++ : курс объектно-ориентированного программирования / У. Савитч. – М. : Вильямс, 2001.
- 13 Хомоненко, А. Д. Базы данных : учебник для высш. учеб. заведений / А. Д. Хомоненко, В. М. Цыганков, М. Г. Мальцев ; под ред. проф. А. Д. Хомоненко. – СПб. : Корона-принт, 2003.

Нормативные документы

- 14 ГОСТ 19.701–90 Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.

Содержание

Предисловие	3
1 Цели дипломного проектирования	4
2 Организация дипломного проектирования	4
3 Общие требования к дипломному проектированию	9
3.1 Требования к технической части дипломного проекта	9
3.2 Рекомендуемая структура технической части дипломного проекта	11
3.3 Графическая часть ДП	21
Рекомендуемая литература	38

Учебное издание

ДИПЛОМНОЕ ПРОЕКТИРОВАНИЕ

Методические указания
для учащихся специальности 2-40 01 01
«Программное обеспечение информационных технологий»

Составители:

Бельчик Марина Анатольевна
Василевская Марина Валерьевна
Скудняков Юрий Александрович
Снежкова Ирина Михайловна

Зав. ред.-издат. отд. О. П. Козельская
Редактор Г. Л. Говор
Корректор Н. Г. Михайлова
Компьютерная верстка Н. М. Олейник

План издания 2007 г. (поз. 3)

Изд. лиц. № 02330/0131735 от 17.02.2004.

Подписано в печать 15.01.2008. Формат 60×84 ¹/₁₆.

Бумага писчая. Гарнитура Таймс. Печать ризографическая.

Усл. печ. л. 2,32. Уч.-изд. л. 2,05. Тираж 150 экз. Заказ 12.

Издатель и полиграфическое исполнение Учреждение образования
«Минский государственный высший радиотехнический колледж»
220005, г. Минск, пр-т Независимости, 62.

