**Основная часть дипломного проекта для специальности 1-40 01 03**

**«Программное обеспечение информационных технологий»**

Дипломный проект это инженерная разработка, выполнив которую дипломник доказывает, что он за период обучения в университете получил знания и овладел навыками выполнения работ в области создания программных средств. Спроектированное программное средство (ПС) может иметь ограниченное число функций, но должно быть обосновано предложенное техническое решение и доказана работоспособность ПС.

Ниже представлены варианты представления основной части дипломного проекта, которые отражают многообразие технологий, используемых при создании программных средств в настоящее время, и решаемых задач.

**Конкретные названия и содержание разделов дипломного проекта, типы и названия чертежей и плакатов дипломник согласовывает с руководителем проекта.**

**Вариант 1**

Под дипломным проектом понимается комплект документации, на основе которого может производится (на промышленном предприятии) изделие. В данном случае, это комплект документации, который позволит получить функционирующее программное средство.

Это значит, что если взять исходные тексты разработанного программного средства (ПС), произвести их компиляцию и компоновку в соответствии с приведенной в пояснительной записке инструкцией, а затем установить (также в соответствии с приведенной в пояснительной записке инструкцией) полученный исполняемый файл (файлы) на компьютере (компьютерах) потребителя, то получим функционирующую, в соответствии с требованиями технического задания, программу.

**1 Анализ прототипов, литературных источников и формирование требований к проектируемому программному средству** (до 15-20 стр.)

Данный раздел должен содержать обзор литературы по теме дипломного проекта, примеры решения аналогичных задач, анализ достоинств и недостатков известных решений. Должны быть рассмотрены не мене 10-15 литературных источников (книги, статьи в журналах, материалы, тезисы и доклады научно-технических конференций, материалы фирм и компаний, научно-технические отчеты, материалы реферативных журналов, патенты, диссертации, стандарты, электронные документы). В списке использованных источников должны быть перечислены рассмотренные материалы, а в тексте раздела содержаться ссылки на них. Раздел может называться в соответствии с темой дипломного проекта. Например, «Анализ принципов организации и функциональных возможностей систем обработки производственных данных машиностроительных предприятий и требования к проектируемому ПС».

На основе проведенного анализа и с учетом требований, указанных в задании на дипломное проектирование, формулируются требования (фактически техническое задание) к проектируемому программному средству, которые должны включать:

а) назначение разработки;

б) состав выполняемых функций;

в) входные данные;

г) выходные данные;

д) требования к временным характеристикам;

е) требования к надежности;

ж) условия эксплуатации;

и) требования к составу и параметрам технических и программных средств;

к) требования к информационной и программной совместимости;

л) обоснование выбора языка и сред разработки;

м) другие требования, имеющие существенное значение для данного проекта.

Пункты а), б), в), г), и), к) требований являются обязательными, остальные требования указываются при необходимости.

**2** **Анализ требований к ПС и** **разработка функциональных требований** (до 15 стр.)

Раздел также может иметь название «Моделирование предметной области и разработка функциональных требований».

В результате работы над этим разделом должны быть сформулированы функциональные требования для проектирования программного средства.

Данный раздел может содержать следующие подразделы.

2.1 Теоретический анализ, математическое обоснование и доказательства, модели технических объектов и результаты моделирования. Данный подраздел не является обязательным.

2.2 Описание функциональности ПС.

Производится с помощью UML-диаграмм, например диаграммы вариантов использования (Use Case или прецеденты). Варианты использования это - описание последовательности действий, которые может осуществлять ПС (система) в ответ на внешние воздействия пользователей или других программных систем. Варианты использования отражают функциональность ПС (системы) с точки зрения получения значимого результата для пользователя.

Описание функциональности также может быть выполнено в виде IDEF – диаграмм.

Если в ДП предполагается разработка БД, то в данном разделе должна быть разработана информационная модель предметной области (например, на языке IDEF1X).

Описание функциональности производится на основе технического задания, разработанного в первом разделе.

2.3 Спецификация функциональных требований.

Должна быть представлена детализация функций проектируемого ПС, которые должны обеспечить реализацию требуемых функций, определенных в техническом задании, с учетом требований к входным и выходным данным, производительности, надежности, техническим, информационным и программным средствам и других требований.

Правильность реализации функции в последующем должна быть проверена с помощью специально разработанных тестов.

***Пример.***

Среди функциональных требований есть такое «Кардиограмма сердечной деятельности человека должна быть представлена в виде графика»

Спецификация данной функции может иметь такой вид.

1. Кардиограмма сердечной деятельности человека представляется в виде графика.

2. Поле для отображения графика масштабируется в соответствии с установленной разрешающей способностью графического адаптера.

3. Единица измерения оси абсцисс – время в секундах.

4. Единица измерения оси ординат – напряжение в мВ.

5. Пользователь должен иметь возможность выбрать цвет фона графика, цвет линии графика, цвет осей графика.

6. Пользователь должен иметь возможность задать шаг для оцифровки осей в единицах времени (ось абсцисс) или напряжения (ось ординат).

7. Пользователь должен иметь возможность задать верхний предел шкалы оси ординат.

8. Пользователь должен иметь возможность управлять значением верхнего предела шкалы оси ординат при просмотре графиков, при этом может задаваться или численное значение верхнего предела шкалы или задаваться шаг изменения верхнего предела шкалы.

9. В распоряжение пользователя должен иметься графический курсор, в виде вертикальной линии. Положение курсора на графике управляется с помощью манипулятора «мышь» или клавиш клавиатуры «стрелка влево», «стрелка вправо».

10. Для каждого положения курсора на графике должны выводится значения времени и напряжения. При отображении значения времени должны выводится две цифры после запятой. При отображении амплитуды должна выводится одна цифра после запятой.

11. Должна быть предусмотрена возможность движения окна графика по временной реализации кардиосигнала.

12. Должна быть предусмотрена возможность экспортирования отображенного графика в офисные приложения Windows.

13. Должна быть предусмотрена возможность отображенного графика в черно-белом изображении на устройстве печати.

14. Должна быть предусмотрена функция вычисления амплитудного спектра исследуемого сигнала. При этом задается число временных точек, для которых вычисляется спектр. Число этих точек должно быть кратно степени двойки.

15. Вычисленный спектр представляется в виде таблицы из двух колонок. в одной колонке выводятся значения частот, в другой колонке выводятся значения амплитуд.

и т.д.

**3** **Проектирование программного средства** (до 15 стр.)

Этот раздел является базовым в дипломном проекте.

В нем должны быть представлены.

3.1 Разработка архитектуры ПС.

Архитектура программного обеспечения ( *software(application) architecture*) − это [структура](http://ru.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0) [программы](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) или вычислительной [системы](http://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0), которая включает программные компоненты, видимые снаружи свойства этих компонентов, а также отношения между ними.

Должны быть определены внутренние и внешние интерфейсы каждой программной составной части.

Для представления архитектуры могут быть использованы:

а) [Диаграмма компонентов](http://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%BE%D0%B2) (*сomponent diagram*) − статическая структурная диаграмма, которая показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами. В качестве физических компонент могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты и т. п.

б) Диаграмма развертывания (синоним − диаграмма размещения). Она применяется для представления общей конфигурации и топологии распределенной программной системы и содержит распределение компонентов по отдельным узлам системы. Кроме того, диаграмма развертывания показывает наличие физических соединений−маршрутов передачи информации между аппаратными устройствами, задействованными в реализации системы. Диаграмма развертывания предназначена для визуализации элементов и компонентов программы, существующих лишь на этапе ее исполнения (runtime). При этом представляются только компоненты-экземпляры программы, являющиеся исполнимыми файлами или динамическими библиотеками. Те компоненты, которые не используются на этапе исполнения, на диаграмме развертывания не показываются.

Все элементы диаграмм должны быть описаны в пояснительной записке.

Архитектура проектируемого ПС может также представляться в виде рисунков их описания.

Разработка архитектуры – творческий процесс. Есть много распространенных способов разработки программных модулей и их связей, в том числе:

− Blackboard;

− Клиент-серверная модель (client-server);

− Архитектуры, построенные вокруг базы данных (database-centric architecture);

− Распределенные вычисления (distributed computing);

− Событийная архитектура (event-driven architecture);

− Front end and back end;

− Неявные вызовы (implicit invocations);

− Монолитное приложение (monolithic application);

− Peer-to-peer;

− Пайпы и фильтры (pipes and filters)

− Plugin;

− Representational State Transfer;

− Rule evaluation;

− Поиск-ориентированная архитектура;

− Сервис-ориентированная архитектура;

− Shared nothing architecture;

− Software componentry;

− Space based architecture;

− Структурированная;

− Трех-уровневая.

Архитектура ПС представляется на плакатах.

3.2 Если разрабатывается база данных, тот производится разработка логической и физической модели базы данных. Эти модели представляются на плакатах.

3.3 Разработка алгоритма ПС и алгоритмов отдельных модулей.

Обобщенный алгоритм ПС представляется схемой программы (согласно ГОСТ 19.701-90). Алгоритмы отдельных модулей представляются схемами алгоритмов или схемами программ.

Разработка алгоритмов, при необходимости, сопровождается их теоретическим обоснованием, моделированием (аналитическим или имитационным), доказательством, что предлагаемый алгоритм обеспечивает решение требуемой задачи.

На этом этапе также могут быть разработаны схемы данных, схемы взаимодействия программ, схемы ресурсов системы (согласно ГОСТ 19.701-90).

Разработанные схемы (некоторые из них, но схема программы обязательно) представляются на чертежах. Должно быть не менее трех чертежей.

Все схемы и алгоритмы должны быть подробно описаны.

**4** **Создание** (**конструирование) программного средства** (до 15 стр.)

На этом этапе выполняется разработка программ, реализующих предложенное техническое решение, и сборка (комплексирование) программного средства.

При необходимости, уточняется выбор языка программирования и средств разработки.

Разрабатываются программные интерфейсы связей между классами, методами, функциями.

Разрабатывается диаграмма классов или структура отдельных модулей. Дается подробное описание классов, атрибутов и методов.

Диаграмма классов должна быть представлена на плакате.

Выполняется программирование (создаются тексты программ) и отладка отдельных модулей проекта. **Тексты программ (основные фрагменты) приводятся в приложении(ях) к пояснительной записке.** Текст программы должен быть подробно документирован.

Производится сборка проекта и комплексная отладка. В тексте пояснительной записки приводится инструкция по сборке ПС.

Приводится описание интерфейсов методов классов или процедур и функций модулей.

В результате выполнения данного этапа должно быть создано функционирующее ПС.

В некоторых проектах, по согласованию с руководителем, разделы 3 и 4 могут быть объединены.

**5** **Тестирование, проверка работоспособности и анализ полученных результатов** (до 10 стр.)

На этом этапе должны быть представлены доказательства того, что спроектированное ПС работает в соответствии с требованиями ТЗ.

Описание тестов, результаты тестирования и другие факты, подтверждающие работоспособность спроектированного ПС представляются в пояснительной записке.

Реально раздел будет содержать некоторое ограниченное число разработанных тестов для проверки работоспособности ПС или некоторой его части, результаты выполненного тестирования, анализ результатов тестирования, а также некоторые экспериментальные проверки на реальных данных.

**6** **Руководство по установке и использованию** (до 10 стр.)

Раздел также может иметь название «Методика использования программного средства».

В данном разделе приводится инструкция (сведения) по установке спроектированного программного средства на компьютере (компьютерах) потребителя. Указываются требуемые аппаратные средства (основные и дополнительные), операционные системы, фреймворки, библиотеки, плагины и т.п.

Руководство (описание) по использованию должно содержать описание действий пользователя при эксплуатации ПС: действия по формированию запросов или входных данных и формы представления ответных результатов или данных.

В этом разделе могут быть представлены примеры и результаты практического применения разработанного ПС и анализ полученных результатов.

В **приложении**(ях) приводится часть исходных кодов разработанного ПС в пределах 30-40 страниц. При необходимости, в приложении могут быть представлены какие-то объемные, но не основные элементы проектирования ПС.

**Примечание.**

Если разрабатываемый проект является инновационной разработкой (решается новая техническая или научно-техническая задача), то второй раздел должен содержать **научно-техническое обоснование предлагаемых решений**:

а) теоретическое обоснование предлагаемых способов и алгоритмов;

б) модели и результаты моделирования исследуемых явлений, систем или устройств;

в) методики достижения поставленных целей;

г) эвристические гипотезы и предложения.

Раздел завершается выводами и уточнением требований к проектируемому ПС. Остальные разделы пояснительной записки остаются без изменений, но их нумерация смещается.

**Вариант 2**

**1** **Анализ литературы по теме дипломного проекта** (15 – 20 страниц)

Раздел должен содержать анализ методов, способов, подходов, методик и т.п., а также анализ существующих аналогов и прототипов разрабатываемого ПС с выделением их достоинств и недостатков. На базе проанализированных недостатков прототипов и требований задания на дипломное проектирование (2-я – 3-я страницы записки) разрабатывается укрупненная спецификация требований, содержащая основные функциональные требования и нефункциональные требования к разрабатываемому ПС, в том числе:

а) назначение разработки;

б) перечень основных выполняемых функций;

в) входные данные;

г) выходные данные;

д) требования к временным характеристикам (при необходимости);

е) требования к надежности (при необходимости);

ж) среда эксплуатации (требования к составу и параметрам технических и программных средств);

и) требования к информационной и программной совместимости (при необходимости);

к) обоснование выбора языка и сред разработки;

л) иные требования (при необходимости).

**2** **Моделирование предметной области (или анализ требований к программному средству)** (10 – 15 страниц)

Материал раздела должен представлять собой основу для разработки функциональной спецификации.

В разделе должна содержаться разработка функциональных моделей предметной области, представленных на каких-то известных языках моделирования (например, UML, IFEF0, DFD и т.п.).

Если в ДП предполагается разработка базы данных (БД), то в данном разделе должна быть разработана информационная модель предметной области (например, на языке IDEF1X).

Раздел может содержать также некоторые теоретические обоснования, математические выкладки, некоторые другие виды моделирования (при необходимости) и т.п.

Раздел должен заканчиваться разработкой функциональной спецификации требований к ПС. В основу данной спецификации должны быть положены основные функциональные требования, приведенные в укрупненной спецификации требований первого раздела, и требования, выявленные по результатам функционального моделирования предметной области. Эта спецификация требований будет являться основой для дальнейшего проектирования.

**3** **Проектирование программного средства** (15 – 20 страниц)

Раздел должен содержать разработку архитектуры ПС и техническое проектирование ПС. Раздел может включать различные виды диаграмм UML, схемы алгоритмов, интерфейсы между компонентами и модулями ПС и т.п., а также их описание.

Раздел должен содержать ссылки на исходные коды, реализующие некоторые из разработанных элементов проекта ПС.

**4** **Тестирование ПС** (5 – 7 страниц)

Раздел должен содержать некоторое ограниченное число разработанных тестов для проверки работоспособности ПС или некоторой его части.

**5** **Методика использования разработанного программного средства** (7 – 10 страниц)

В разделе приводятся основные сведения по работе с ПС.

**Приложения**

Должна быть приведена часть исходных кодов разработанного ПС в пределах 30 – 40 страниц. При необходимости могут быть приведены какие-то второстепенные элементы проектирования ПС.