

Министерство образования и науки Российской Федерации
ФГБОУ ВПО «Тамбовский государственный технический
университет»

Факультет «Магистратура»

И.П. Рак

**МЕТОДОЛОГИЯ И ТЕХНОЛОГИЯ
ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ
СИСТЕМ**

Утверждено Методическим советом ТГТУ
в качестве методических указаний для студентов
магистратуры, обучающихся по направлению 230700.68
«Прикладная информатика»



Тамбов
2013

Рецензент
к.т.н., доцент С.Н. Баршутин

Методология и технология проектирования информационных систем: Метод. указ. / Сост.: И.П. Рак, Тамбов: ТГТУ, 2013. – 16 с.

Утверждено Методическим советом ТГТУ
(протокол № ____ от _____)

СОДЕРЖАНИЕ

Введение.....	4
1 Жизненный цикл информационных систем.....	6
2 Методические аспекты проектирования информационных систем.....	8
3 Проектирование архитектуры информационных систем.....	9
4 Характеристики качества информационных систем.....	1
5 Верификация, тестирование и испытание информационных систем.....	1
6 Управление разработкой информационных систем.....	1
Список рекомендуемых источников.....	4
	1
	5
	1
	6

ВВЕДЕНИЕ

Дисциплина «Методология и технология проектирования информационных систем» направлена на формирование у обучающихся теоретических знаний о современных методологиях, методах и средствах разработки информационных систем.

Изучение дисциплины служит формированию следующих компетенций:

общекультурные компетенции (ОК):

- способен приобретать и использовать на практике знания, умения и навыки в организации исследовательских и проектных работ, в управлении коллективом (ОК-3);
- способен проявлять инициативу, брать на себя ответственность в условиях риска и принимать нестандартные решения в проблемных ситуациях (ОК-4);
- способен управлять знаниями в условиях формирования и развития информационного общества: анализировать, синтезировать и критически резюмировать и представлять информацию (ОК-6);

профессиональные компетенции (ПК):

- способен на практике применять новые научные принципы и методы исследований (ПК-3);
- способен к профессиональной эксплуатации современного электронного оборудования в соответствии с целями ООП магистратуры (ПК-4);
- способен проводить научные эксперименты, оценивать результаты исследований (ПК-8);
- способен выбирать методологию и технологию проектирования ИС с учетом проектных рисков (ПК-11);
- способен анализировать данные и оценивать требуемые знания для решения нестандартных задач с использованием математических методов и методов компьютерного моделирования (ПК-12);
- способен анализировать и оптимизировать прикладные и информационные процессы (ПК-13);
- способен применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов и создания ИС (ПК-15);
- способен проектировать архитектуру и сервисы информационных систем предприятий и организаций в прикладной области (ПК-16);

- способен проектировать информационные процессы и системы с использованием инновационных инструментальных средств, адаптировать современные ИКТ к задачам прикладных ИС (ПК-17);
- способен принимать эффективные проектные решения в условиях неопределенности и риска (ПК-18);
- способен использовать международные информационные ресурсы и стандарты в информатизации предприятий и организаций (ПК-26);
- способен использовать информационные сервисы для автоматизации прикладных и информационных процессов (ПК-27);
- способен интегрировать компоненты и сервисы информационных систем (ПК-28).

1 ЖИЗНЕННЫЙ ЦИКЛ ИНФОРМАЦИОННЫХ СИСТЕМ

Рассматриваемые вопросы:

- Этапы развития технологий разработки программного обеспечения.
- Системные основы современных технологий разработки программного обеспечения.
- Стандарт жизненного цикла.
- Профили стандартов жизненного цикла.
- Процессы жизненного цикла.
- Модели жизненного цикла.

Этапы развития технологий разработки программного обеспечения. Индустрия программного обеспечения (ПО) начала зарождаться в середине 50-х годов прошлого столетия. Однако до конца 60-х ей не уделялось серьёзного внимания, поскольку её доля в компьютерном бизнесе была очень мала. Серьёзный рост начался в 70-х годах. Большой скачок в индустрии произошёл после появления персонального компьютера. На сегодня производство ПО является крупнейшей отраслью мировой экономики.

В развитии технологии программирования можно выделить три этапа.

- I. Осмысление опыта разработки больших систем, включающее понимание важности того, как ведётся разработка ПО, а не только на каком языке программирования это делается.
- II. Разработка новых технологических подходов.
- III. Принятие стандартов на состав процессов жизненного цикла ПО и многочисленные попытки решить проблему качества ПО.

Системные основы современных технологий разработки ПО.

Быстрый рост областей применения, сложности функций и масштабов информационных систем (ИС) привело к принципиальному изменению методов в сфере разработки ПО и к переходу от технологии индивидуального программирования отдельных небольших программ к коллективному созданию крупных ИС инженерными методами проектирования и разработки (*software engineering*).

Программная инженерия (software engineering) – это область компьютерной науки, которая занимается построением программных систем, настолько больших и сложных, что для этого требуется участие слаженных команд разработчиков различных специальностей и квалификаций. Суть методологии программной инженерии состоит в применении систематизированного, научного и предсказуемого процесса проектирования, разработки и сопровождения ПО.

Стандарт жизненного цикла ИС. Одним из базовых понятий методологии проектирования ИС является понятие *жизненного цикла* (ЖЦ). ЖЦ ИС – это непрерывный процесс, начинающийся с момента

принятия решения о необходимости создания ИС и заканчивающийся в момент полного её изъятия из эксплуатации.

Основным нормативным документом, регламентирующим ЖЦ ИС, является международный стандарт ISO/IEC 12207 (его российский аналог ГОСТ Р ИСО/МЭК 12207). Он определяет структуру ЖЦ, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания и эксплуатации ПО.

Профили стандартов жизненного цикла. Разработка больших проектов ИС связана с работой коллективов из нескольких организаций. Для их слаженной работы необходима совокупность документов, регламентирующих различные этапы разработки ИС. Комплекс таких документов называют *нормативно-методическим обеспечением* (НМО).

В состав НМО входят стандарты и руководящие документы, методики выполнения сложных операций, шаблоны проектных и программных документов.

Профиль стандартов – это совокупность нескольких базовых стандартов и других нормативных документов с чётко определённым подмножеством обязательных и факультативных возможностей, предназначенная для реализации заданных функций ИС.

Процессы жизненного цикла. В соответствии со стандартом ISO/IEC 12207 все процессы ЖЦ ИС разделены на три группы:

- основные процессы (приобретение, поставка, разработка, эксплуатация, сопровождение);
- вспомогательные процессы, обеспечивающие выполнение основных процессов (документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, совместная оценка, аудит, разрешение проблем);
- организационные процессы (управление проектом, создание инфраструктуры проекта, совершенствование, обучение).

Модели жизненного цикла ИС. Под моделью ЖЦ ИС понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении ЖЦ. Стандарт ISO/IEC 12207 не предлагает конкретную модель ЖЦ ИС. Его положения являются общими для любых моделей ЖЦ, методов и технологий создания ИС.

Модель ЖЦ ИС включает в себя: стадии, результаты выполнения работ на каждой стадии и ключевые события.

Традиционно выделяют следующие основные стадии ЖЦ ИС: разработка требований, проектирование, реализация, тестирование и отладка, эксплуатация и сопровождение.

Наиболее распространёнными моделями ЖЦ являются каскадная и итерационная.

2 МЕТОДИЧЕСКИЕ АСПЕКТЫ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

Рассматриваемые вопросы:

- Общие принципы проектирования систем.
- Основные принципы объектно-ориентированного подхода.
- Методы проектирования ИС.
- Понятие CASE-технологии.

Общие принципы проектирования систем. Система – это совокупность взаимодействующих компонентов, работающих совместно для достижения определенных целей. Свойства и поведение системных компонентов влияют друг на друга сложным образом. Корректное функционирование каждого системного компонента зависит от функционирования многих других компонентов.

Для решения проблемы сложности системы широко используется метод иерархической декомпозиции. Согласно данному подходу сложная система разбивается на более простые части, каждая из которых, в свою очередь, строится из частей меньшего размера, и т.д., до тех пор, пока самые небольшие части можно будет строить из имеющегося материала.

По отношению к проектированию ИС это означает, что её необходимо разделить на небольшие модули (подсистемы), каждый из которых можно разрабатывать независимо от других.

Проектирование модулей включает в себя разработку локальных функций и подробных описаний алгоритмов обработки данных; межмодульных интерфейсов; внутренних структур данных; структурных схем передачи управления; средств управления в исключительных ситуациях.

Основные принципы объектно-ориентированного подхода. Объектно-ориентированный подход использует объектную декомпозицию, при этом статическая структура системы описывается в терминах объектов и связей между ними, а поведение системы – в терминах обмена сообщениями между объектами. Каждый объект системы обладает своим собственным поведением, моделирующим поведение объекта реального мира.

Главное преимущество объектно-ориентированного программирования состоит в том, что оно упрощает задачу внесения изменений в системную архитектуру, поскольку представление состояния объекта не оказывает на неё влияния. Изменение внутренних данных объекта не должно влиять на другие объекты системы. Более того, так как объекты слабо связаны между собой, обычно новые объекты просто вставляются без значительных воздействий на остальные компоненты системы.

Методы проектирования ИС. Методы проектирования ИС можно классифицировать по степени использования средств автоматизации, типовых проектных решений, адаптивности к предполагаемым изменениям.

По степени автоматизации методы проектирования разделяются на: ручное и компьютерное.

По степени использования типовых проектных решений различают оригинальное (индивидуальное) и типовое проектирования.

Типовое проектное решение (ТПР) – это пригодное к многократному использованию проектное решение. ТПР по уровню декомпозиции можно разделить на следующие классы:

- элементные ТПР – типовые решения по задаче или по отдельному виду обеспечения задачи;
- подсистемные ТПР – в качестве элементов типизации выступают отдельные подсистемы;
- объектные ТПР – типовые отраслевые проекты, которые включают полный набор подсистем ИС.

По степени адаптивности проектных решений выделяют методы: реконструкции, параметризации и реструктуризации модели.

Понятие CASE-технологии. Тенденции развития современных информационных технологий приводят к постоянному возрастанию сложности ИС. Для успешной реализации ИС должна быть адекватно описана, должны быть построены полные и непротиворечивые функциональные и информационные модели системы. Это способствовало появлению программно-технологических средств специального класса – *CASE-средств*.

Понятие CASE (Computer Aided Software Engineering) первоначально было ограничено только задачами автоматизации разработки ИС, в настоящее время оно охватывает все процессы ЖЦ ИС.

CASE-технология представляет собой совокупность методов проектирования ИС, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать эту модель на всех стадиях разработки и сопровождения системы и разрабатывать приложения в соответствии с информационными потребностями пользователей.

3 ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ ИНФОРМАЦИОННЫХ СИСТЕМ

Рассматриваемые вопросы:

- Понятие архитектуры ИС.
- Планирование архитектуры.

- Проектирование архитектуры.
- Документирование программной архитектуры.

Понятие архитектуры ИС. Каждая ИС имеет ту или иную архитектуру. Если при разработке ИС её архитектурные представления использовались явно, то это позволяет повысить степень успешности разработок.

Архитектура, в первую очередь, – это совокупность принципов структурирования, которые предоставляют возможность в контексте ИС как целого представить её с помощью набора более простых систем, каждая из которых определена в соответствующем локальном контексте.

Архитектура определяет не только структуры ИС через элементы и связи, но также и взаимодействие элементов структур. В этом плане она представляет поведение ИС в терминах процессов их функционирования.

Конкретная ИС разрабатывается в конкретной среде разработки и для конкретной среды использования. Факторы этих окружений не могут не влиять на архитектурные решения. К числу основных факторов, оказывающих влияние на архитектуру, относятся: назначение ИС; лица, заинтересованные в разработке ИС; ограничения разных типов; уровень квалификации лиц, вовлечённых в процессы использования ИС.

Планирование архитектуры. Современные методы разработки ПО предполагают обратную связь между всеми действующими лицами, от проектировщика до аналитика. Все эти лица являются участниками процесса создания архитектуры ИС.

На создание программной архитектуры направленно ряд операций процесса разработки ПО:

- создание экономической модели системы;
- выявление требований;
- создание новой или выбор существующей архитектуры;
- документирование и распространение сведений об архитектуре;
- анализ или оценка архитектуры;
- реализация системы на основе архитектуры;
- проверка соответствия реализации архитектуре.

Проектирование архитектуры. *Атрибутный метод проектирования* – это методика определения программной архитектуры, в которой процесс декомпозиции основывается на предполагаемых атрибутах качества продукта. Это рекурсивный процесс декомпозиции, на каждом из этапов которого происходит отбор тактик и архитектурных образ-

цов, удовлетворяющих тем или иным сценариям качества, а также распределение функциональности, направленное на конкретизацию типов модулей данного образца.

После проектирования архитектуры и формирования рабочих групп можно приступать к конструированию макета системы. Цель этого этапа в том, чтобы обеспечить основополагающую возможность реализации функциональности системы в предпочтительном порядке.

Документирование программной архитектуры. Характер архитектуры любой системы обуславливается предъявляемыми к ней требованиями, это утверждение справедливо и по отношению к документации архитектуры, другими словами, содержание документации зависит от предполагаемых вариантов её применения. С одной стороны, документация должна быть абстрактной и доступной для понимания новыми сотрудниками, а с другой – весьма детальной – настолько, чтобы её можно было использовать как план проведения анализа. Между архитектурной документацией, предназначенной, например, для проведения анализа безопасности, и архитектурной документацией для изучения конструкторами должны существовать серьёзные различия. С другой стороны, у этих вариантов будет мало общего с содержанием руководства для ознакомления, предназначенного новому сотруднику.

Из всего этого следует, что заинтересованные в составлении документации лица преследуют разные цели – от представленной в ней информации они требуют разной направленности, разных уровней детализации и разных трактовок.

4 ХАРАКТЕРИСТИКИ КАЧЕСТВА ИНФОРМАЦИОННЫХ СИСТЕМ

Рассматриваемые вопросы:

- Свойства и атрибуты качества функциональных возможностей ИС.
- Конструктивные характеристики качества ИС.
- Характеристики качества баз данных.
- Характеристики защиты и безопасности функционирования ИС.
- Выбор характеристик качества в проектах ИС.

Свойства и атрибуты качества функциональных возможностей ИС. Основной характеристикой качества функциональных возможностей ИС является функциональная пригодность системы. *Функциональная пригодность* – это набор и описания атрибутов, определяющих назначение, основные, необходимые и достаточные функции ИС, заданные техническим заданием (ТЗ) и спецификациями требований заказчика. В процессе проектирования системы атрибуты функциональной пригод-

ности должны конкретизироваться в спецификациях на ИС в целом и на компоненты в отдельности. Атрибутами этой характеристики могут быть:

- функциональная полнота решения заданного комплекса задач;
- степень покрытия функциональных требований спецификациями и их стабильность при совершенствовании ИС;
- число реализуемых требований заказчиков и т.д.

Кроме них функциональную пригодность отражают множество различных специализированных критериев, которые тесно связаны с конкретными решаемыми задачами и сферой применения системы. Их можно рассматривать как частные критерии или как факторы, влияющие на основной показатель качества ИС.

В ряде систем особое значение для функциональной пригодности имеет системное проектирование организации информационного обеспечения и баз данных (БД).

Конструктивные характеристики качества ИС. К основным конструктивным характеристикам качества ИС относятся:

Надёжность – это свойства ИС обеспечивать достаточно низкую вероятность потери работоспособности (отказа) в процессе функционирования в реальном времени.

Завершённость – это свойство ИС не попадать в состояния отказов вследствие ошибок и дефектов в программах и данных.

Устойчивость к дефектам и ошибкам – это свойство ИС автоматически поддерживать заданный уровень качества функционирования при проявлениях дефектов и ошибок.

Восстанавливаемость – это свойство ИС в случае отказа возобновлять требуемый уровень качества функционирования, а также исправлять поврежденные программы и данные.

Доступность (готовность) – это свойство ИС быть в состоянии выполнять требуемую функцию в данный момент времени при заданных условиях использования.

Эффективность характеризует динамику функционирования компонентов ИС. Выделяют две характеристики эффективности: временная эффективность и использование ресурсов компьютера.

Практичность (применимость) – это свойства ИС, отражающие сложность её понимания, изучения и использования.

Сопровождаемость – это приспособленность ИС к модификации и изменению конфигурации.

Мобильность – это подготовленность ИС к переносу из одной аппаратно-операционной среды в другую.

Характеристики качества БД. БД можно рассматривать как два компонента:

1. программное средство системы управления БД;
2. данные БД.

Хотя эти компоненты тесно взаимодействуют при реализации конкретной прикладной БД, первоначально при проектировании они создаются или выбираются практически независимо и могут рассматриваться в ЖЦ как два объекта, которые различаются:

- номенклатурой и содержанием показателей качества, определяющих их назначение, функции и потребительские свойства;
- технологией и средствами автоматизации разработки и обеспечения всего ЖЦ каждого объекта;
- категориями специалистов, обеспечивающих: создание, эксплуатацию и применение компонентов БД;
- комплектами эксплуатационной и технологической документации, поддерживающими ЖЦ объектов.

Характеристики защиты и безопасности функционирования ИС.

Непрерывно возрастающая сложность и вследствие этого уязвимость систем от случайных и преднамеренных негативных воздействий выдвинули проблему безопасности в разряд важнейших. При этом выделяют:

1. *информационную безопасность*, связанную в основном с защитой от преднамеренных, негативных воздействий на информационные ресурсы систем;
2. *функциональную безопасность*, обусловленную отказовыми ситуациями и потерей работоспособности систем вследствие проявления непреднамеренных, случайных дефектов и отказов программ, данных и аппаратуры.

Эффективная система защиты данных и ИС подразумевает наличие совокупности организационных и технических мероприятий, направленных на предупреждение различных угроз безопасности, их выявление, локализацию и ликвидацию. Создание такой системы предусматривает планирование и реализацию целенаправленной политики комплексного обеспечения безопасности системы.

Выбор характеристик качества в проектах ИС. Стандарт ISO 9126 не содержит рекомендаций и методик выбора значений характеристик качества ИС для конкретного проекта. Необходимо установить рациональные диапазоны мер и шкал для каждой характеристики и её атрибутов, которые можно будет использовать в качестве первичных ограничений при выборе их значений для реальных проектов. Далее должны быть разработаны процессы выбора, установления и представления в спецификациях требований к атрибутам каждой характеристики качества. Эти требования должны учитывать реальные ограничения ресурсов, доступных для обеспечения ЖЦ ИС. Ресурсы этих процессов и атрибуты характеристик качества далее сводятся к трудоёмкости и длительности их реализации.

5 ВЕРИФИКАЦИЯ, ТЕСТИРОВАНИЕ И ИСПЫТАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

Рассматриваемые вопросы:

- Принципы верификации и тестирования ИС.
- Технологические этапы и стратегии систематического тестирования ИС.
- Процессы оценивания характеристик и испытания ИС.
- Средства для испытаний и определения характеристик сложных ИС.

Принципы верификации и тестирования ИС. *Верификация* – это процесс для определения, выполняет ли ИС и её компоненты требования, наложенные на них в этапах ЖЦ ИС. Основная цель верификации ИС состоит в том, чтобы обнаружить, зарегистрировать и устранить дефекты и ошибки, которые внесены во время её разработки и модификации.

Цели верификации ИС достигаются посредством последовательного выполнения комбинации из просмотров, анализов, разработки тестовых сценариев и процедур и последующего их выполнения.

Результаты верификации должны быть включены в документы: выполненные процедуры верификации ИС, описание и отчёт о квалификации ИС и её компонентов.

Технологические этапы и стратегии систематического тестирования ИС. Исходным эталоном для тестирования любой ИС являются требования ТЗ и спецификации требований заказчика к создаваемой системе. Подобные документы должны устанавливать состав, содержание и значения результатов, которые должен получать пользователь при определённых условиях и исходных данных. Любое отклонение результатов функционирования ИС от предъявляемых к ней требований следует квалифицировать как ошибку или дефект.

При *нисходящем тестировании от требований* оно начинается с программ организации вычислительного процесса. Первоначально тестируются управляющее ядро системы и программные компоненты, решающие функциональные задачи, размещённые на высших иерархических уровнях, на соответствие исходным требованиям ТЗ. К ним последовательно, по мере готовности, подключаются компоненты более низких иерархических уровней. Если некоторые программные компоненты нижних уровней не разработаны или недостаточно протестированы, то вместо них временно могут подключаться программные имитаторы – "заглушки".

При систематическом *восходящем тестировании*, прежде всего, проверяются программные компоненты и модули нижних иерархических уровней, к которым последовательно подключаются вызывающие

их модули. В этих модулях тестирование также начинается с простейших конструкций, переменных и маршрутов обработки данных.

Процессы оценивания характеристик и испытания ИС. Для оценивания характеристик и испытаний ИС на различных этапах ЖЦ в качестве методологической основы целесообразно использовать рекомендации стандарта ISO 14598. Процесс оценивания ИС в стандарте представлен как совокупность действий, выполняемых совместно заказчиком и оценщиком (испытателем). Потенциальными заказчиками оценивания ИС могут быть разработчики, поставщики, пользователи, производители ИС, а также третейские испытательные лаборатории программных продуктов.

Характеристики качества функционирования ИС зависят не только от её внутренних свойств, но и от свойств внешней среды, в которой они применяются. Для сокращения неопределённостей и ошибок при оценивании качества ИС необходимо до начала испытаний определить основные параметры внешней среды, при которых она должна функционировать при испытании и эксплуатации.

Средства для испытаний и определения характеристик сложных ИС. Для обеспечения высокого качества крупных ИС необходимы соответствующие проблемно-ориентированные интегрированные системы автоматизации тестирования, способные достаточно полно заменить испытания программ с реальными объектами внешней среды. При этом высокая стоимость и риск испытаний с реальными объектами почти всегда оправдывают значительные затраты на такие интегрированные системы, если предстоят испытания критического ИС с высокими требованиями к качеству функционирования, с длительным ЖЦ и множеством развивающихся версий.

В отличие от натурального эксперимента моделирование внешней среды и тестов на компьютере имеет большие возможности контроля как исходных данных, так и всех промежуточных и выходных результатов функционирования испытываемого объекта.

6 УПРАВЛЕНИЕ РАЗРАБОТКОЙ ИНФОРМАЦИОННЫХ СИСТЕМ

С достаточно общих позиций можно считать задачей управления проектами эффективное использование ресурсов для достижения нужных результатов.

Основные виды ресурсов, используемых в любом проекте: время; бюджет; персонал; используемое оборудование, инструменты, материалы, и т.п.

Виды деятельности, которыми приходится заниматься руководителю проекта или группе управления проектом для обеспечения его успешного выполнения, можно разделить на следующие области:

- управление содержанием проекта и качеством;
- управление ресурсами проекта;
- управление рисками;
- управление коммуникациями и информационное обеспечение проекта;
- управление конфигурациями и изменениями;
- управление проектной средой и технологиями;
- контроль и мониторинг состояния проекта.

СПИСОК РЕКОМЕНДУЕМЫХ ИСТОЧНИКОВ

1. Батоврин В.К. Системная и программная инженерия. Словарь-справочник: учеб. пособие для вузов / В. К. Батоврин. – СПб.: ДМК Пресс, 2010. – 280 с.
2. Брайант Р.Э. Компьютерные системы. Архитектура и программирование. Взгляд программиста / Р. Э. Брайант, Д. Р. О'Халларон. – СПб.: БХВ-Петербург, 2005. – 1104 с.
3. Введение в программную инженерию: Учебный курс / Кознов Д.В. и [др.]. – СПб.: Санкт-Петербургский государственный университет, 2009. – 154 с.
4. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: учебник для вузов / А. М. Вендров. – М.: Финансы и статистика, 2005. – 544 с.
5. Гагарина Л.Г. Основы технологии разработки программных продуктов: учебное пособие для сред. проф. образования / Л. Г. Гагарина, Б. Д. Виснадул, А. В. Игошин. – М.: ФОРУМ, 2006. – 192 с.
6. Гецци К. Основы инженерии программного обеспечения: пер. с англ. / К. Гецци, М. Джазайери, Д. Мандриоли. – СПб.: БХВ-Петербург, 2005. – 832 с.
7. Иванова Г.С. Технология программирования: учебник для вузов / Г. С. Иванова. – М.: Изд-во МГТУ им. Н.Э.Баумана, 2006. – 336 с.
8. Ипатова Э.Р. Методологии и технологии системного проектирования информационных систем: учебник / Э.Р. Ипатова, Ю.В. Ипатов. – М.: Флинта; МПСИ, 2008. – 256 с.