

п р и к л а д н а я ИНФОРМАТИК@

научно-практический
журнал

№ 1 (43) 2013

Январь-февраль

ISSN 1993-8314

Московский финансово-промышленный университет «Синергия»

С 19 февраля 2010 года журнал включен в Перечень ведущих периодических изданий, рекомендованных ВАК для публикации результатов диссертационных исследований.

Уважаемые коллеги!

8–12 апреля состоится значимое научное мероприятие на базе Московского финансово-промышленного университета «Синергия» — VIII Международный научный конгресс «Роль бизнеса в трансформации общества». Журнал «Прикладная информатика» принимает непосредственное участие в его организации, проведении и освещении итогов. Подробнее о мероприятии — на второй странице обложки. Обращаем внимание читателей и подписчиков на следующие тематические подсекции, которые будут представлены на конгрессе:

- Информационные технологии как точка роста современной экономики.
- Программное и аппаратное обеспечение вычислительных систем.
- Облачные технологии и большие данные.
- Основные тенденции развития электронного обучения.

Состав рубрик данного номера традиционный. Статьи отражают современные научные направления и предлагают практические решения.

Актуальная масштабная публикация практического характера «Подход к комплексному применению методологий систематизации требований» представлена А. В. Симкиным, ведущим консультантом компании «ИБС Софт».

Авторы из Нижегородского государственного университета им. Н. И. Лобачевского профессор Ю. А. Кузнецов и старший преподаватель О. В. Мичасова опубликовали новые научные результаты в работе «Численно-аналитическое исследование модели экономического роста Лукаса».

Одна из ключевых рубрик журнала «Simulation» представлена описанием концепции и возможностей нового направления в компьютерном моделировании и соответствующего программного продукта: статья А. А. Емельянова «Концепция и возможности акторно-ориентированной системы имитационного моделирования “Actor Pilgrim”» (продолжение. Начало в ПИ №6 (42). Материал публикуется по просьбам наших читателей — участников НП «Национальное общество имитационного моделирования».

Главный редактор
А. А. Емельянов

А. В. Симкин, ведущий консультант ООО «ИБС Софт», г. Москва

Подход к комплексному применению методологий систематизации требований

В статье описывается подход, отражающий комплексное применение различных методологий для системной формализации требований. Он описывает применение рамочной модели и может быть использован для решения задач систематизации требований, возникающих при проектировании информационной системы. Решение таких задач зачастую связано с определением классификации и взаимосвязей между различными видами требований.

Введение

Несмотря на то что описанный далее подход представляет собой определенный набор хорошо известных методов из различных методологий, главная цель его применения — создание простого и понятного документа, отражающего взгляды заинтересованных лиц всех уровней (рис. 1).

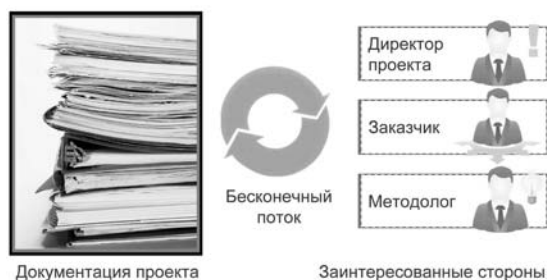


Рис. 1. Бесконечный поток документации для заинтересованных лиц

Данный подход возник в результате диссертационного исследования методологий создания информационно-аналитических систем, а также апробации полученных результатов в ходе работ по проектированию и внедрению различных информационных систем. В качестве исходных данных для применения настоящего подхода необходимо наличие множества неструктурированной и даже неформализованной информации, которую необходимо преобразовать в набор технических требований к информационной

системе. Процесс преобразования назовем систематизацией, ее правила описываются определенной рамочной моделью (далее — модель систематизации требований). Под систематизацией понимается объединение или сведение различных групп требований по неким признакам, параметрам или критериям в единую иерархию функциональных целей согласно определенным взаимосвязям между внутренними и внешними требованиями. Систематизация требований описывается с помощью:

- определения взаимосвязей между составом требований к информационной системе;
- некоторой иерархии групп требований, их элементов и детализированных спецификаций, определяемых едиными принципами классификации.

В статье не только описан структурированный и последовательный подход к применению указанной модели, но и отражен практический опыт его применения к созданию документов подобного уровня при проектировании информационной системы. Несмотря на то что будет описана лишь часть из множества различных существующих наборов требований, это несколько не ограничивает их возможный состав.

Итак, далее в статье будут отражены три тезиса.

1. Модель систематизации требований не должна быть ограничена набором конкретных методов и инструментов, а лишь от-

ражает возможные правила их использования с точки зрения профессиональной позиции для каждой заинтересованной стороны.

2. Классификация и кодирование требований, имеющих однозначные взаимосвязи на всех уровнях в иерархии своих описаний, разрешает проблемы информационного характера, связанные со сложностью (снижает ее) восприятия и работы с множеством требований к ИС.

3. Набор групп требований определяется в зависимости от концепции самой информационной системы и ее будущего содержания.

1. Систематизация требований

1.1. Необходимость решения поставленной задачи

Проектирование информационной системы (далее — ИС) — сложная комплексная задача, требующая не только учета огромного количества предъявляемых требований со стороны заинтересованных сторон (англ. *stakeholders*), но и их гармонизации, выявления совпадений и несоответствий. Она может быть решена как одним человеком, так и требовать работы не только одной проектной команды, но и небольшого «производства». Возникающие сложности информационного характера [1] можно разрешить, применяя системный подход.

Системный подход возник достаточно давно, в его основе лежит попытка разрешения информационных сложностей. Уоррен Уивер, один из основателей теории информации, утверждал, что «в современной физике и биологии повсюду возникают проблемы организованной сложности, то есть взаимодействия большого, но не бесконечного числа переменных, и они требуют новых понятийных средств для своего разрешения» [2]. Эти понятийные средства в окончательном виде сформулировал Л. Фон Берталанфи, назвав «общей теорией систем» [2]. Идеология оказалась настолько мощной, что ее тут же принялись активно применять во всех

областях знаний, а кибернетика использовала данную теорию как основополагающую. Методология системного анализа, получившая значительное развитие уже в эпоху компьютерной техники, позволила обобщить существующие методы и инструменты, определив «научный метод познания, представляющий собой последовательность действий по установлению структурных связей между переменными или элементами исследуемой системы» [3]. С основами системного подхода можно ознакомиться, например, в [4].

Таким образом, поняв исключительную необходимость систематизации требований, перейдем к определению целей и задач формируемой системы требований.

1.2. Цели и задачи формализации требований

Теория систем определяет истинное значение постановки цели (целей) и задач функционирования любых компонентов, элементов, методов и т. д., относящихся к некоторой рассматриваемой совокупности. Формализация технических требований к информационной системе позволяет выстроить их в единую систему в виде группы взаимосвязанных требований (или даже множества групп). В данном контексте можем выделить три ключевые задачи формализации требований (рис. 2).



Рис. 2. Задачи формализации технических требований

Поставленные выше задачи не ограничивают рамки реализации какого-то конкретного проекта. Они каждый раз определя-

ются отдельно в зависимости от контекста и особенностей самого проекта, а также того, для чего именно необходимо сводить все требования в единую систему.

Таким образом, после того как достигнуто понимание цели (целей) и задач, следует определить, сформулировать и описать последовательность шагов для их достижения.

1.3. Способы решения поставленных задач

На этом этапе воспользуемся одним из методов системного анализа, а именно декомпозицией элементов. Для каждой из поставленных задач найдем способы их решения, помогающие дать ответ на вопрос: что необходимо описать, чтобы решить поставленную задачу?

Например, для задачи формализации технических требований может быть получена следующая декомпозиция (рис. 3):

1. *Верхний блок* — обеспечение единого видения проектируемой ИС у всех заинтересованных лиц.

2. *Средний блок* — формирование проектных решений и управление требованиями.

3. *Нижний блок* — разработка и проведение тестирования и приемки системы.



Рис. 3. Способы решения поставленных задач

После нахождения способов решения задач необходимо обозначить набор методов и инструментов для их решения. Выбор всегда должен определяться целесообразностью применения данного набора. Золотым правилом в этом случае является следующий принцип: «Использование всего лучшего, что накоплено различными методика-

ми, поэтому важно понимать в общих чертах их сильные и слабые стороны» [5].

На рисунке 4 представлен пример результата выбора методов и инструментов в рамках ранее поставленных задач.



Рис. 4. Методы и инструменты решения поставленных задач

1.4. Определение подхода систематизации

Подведем итоги первоочередного определения задачи систематизации требований. Для начала при использовании системного подхода необходимо выявить цели и задачи не только системы в целом, но и любых ее элементов (в данном случае это документы, содержащие набор технических требований). Следующим шагом является построение структурной декомпозиции от уровня сформулированных задач до конкретных методов и инструментов, которые будут использоваться при формализации требований. Таким образом, сформировав уникальную структурную декомпозицию целей и задач, необходимо определить состав шагов (этапов) процесса ее реализации.

Далее установим правила систематизации требований, описанные в виде эталонной модели, позволяющей отталкиваться от нее при поиске уникального пути решения той или иной задачи. В следующих разделах последовательно рассмотрим составляющие данной модели:

- описание модели систематизации требований и примеры ее применения;
- принципы взаимосвязи, классификации и кодирования требований и их описание;
- спецификации требований.

2. Модель систематизации требований

2.1. Концепция модели

Формирование эталонной модели систематизации требований позволяет определить рамочную модель или фреймворк (англ. *framework*) для использования при формализации и систематизации технических требований к информационной системе. Фреймворк, например, может представлять собой структуру (шаблон) документа, в котором будет отражаться результат формализации требований, а также набор принципов (подходов) к их систематизации. Рассмотрим подробнее данную модель систематизации (рис. 5) с целью понимания способов ее реализации.

На концептуальном уровне модель представляет собой иерархию уровней абстракции (англ. *views*) с точки зрения профессиональной позиции каждой заинтересованной стороны. Дж. Захман в своей архитектурной онтологии называет эти уровни *Reification Transformations* [6]. В случае рассмотрения архитектурной составляющей информационной системы (особенно на низких уровнях абстракции) можно определить соответствия с онтологией представлений архитектуры информационной системы (*viewpoints*) [7].



Рис. 5. Модель систематизации требований

2.2. Описание уровней модели

Главная задача описанных далее уровней — обеспечение сквозного и однознач-

ного понимания целей и задач, поставленных на уровне владельца бизнеса (руководителя), вплоть до построения конечной технологической модели, отражающейся в логическом программном коде исполняемых файлов, обеспечивающих непрерывность бизнеса. Рассмотрим уровни подробнее.

Концептуальный уровень: взгляд внешнего заказчика в лице владельца бизнеса (*business scope planners*), менеджера процесса (англ. *business concept owners*) и бизнес-аналитика (англ. *business analyst*).

Описывается назначение и цели создания системы, а также бизнес-логика. Бизнес-логика — это реализация предметной области в информационной системе, которая может иметь различные виды представлений. В общем случае это целостное описание характеристик объекта автоматизации, включающее в себя:

- общие сведения об объекте автоматизации;
- описание комплексов задач;
- функциональные компоненты системы (выполняемые функции).

Данные описания могут представляться в различных формах, в том числе в виде модели бизнес-процессов (подробнее см. разделы 3.1 и 3.2).

Архитектурный уровень: взгляд системного архитектора (англ. *system architect*), методолога.

Представлена архитектура приложений информационной системы, представляющая собой обеспечение потребностей автоматизируемых бизнес-процессов набором функциональных компонентов (функций) системы [5]. Практическая реализация этого описания может иметь различные представления. Самый простой вариант (см. раздел 4) — перечень групп требований (англ. *types of requirements*). Группа требований — это логическое объединение однотипных требований, т. е. результат объединения их в единую систему [8]. Обобщением в рамках данного описания, а также приведением его к ГОСТу 34 «Стандарты на раз-

работку автоматизированных систем» является обозначение уровня классификации, определяющего, к какому виду обеспечения и к какой составляющей системы относится та или иная группа требований (декомпозиция комплексов задач на функции).

Системный уровень: взгляд системного аналитика (англ. *systemanalyst*).

Уровень отражает логическое описание работы информационной системы, а именно перечень требований и их взаимосвязь (см. раздел 5). При некотором приближении можно назвать этот уровень системным проектом. Системный проект — это общее техническое представление информационной системы, он включает в себя полноценную функциональную модель, информационную модель и технические требования. Кроме того, в данном разделе может определяться объем и организация работ, требуемые ресурсы и проектные риски.

Технологический уровень: взгляд разработчика (англ. *developer*).

Описывается технологическая модель системы в виде спецификаций для програм-

миста. Спецификация определяет, что и каким образом должна делать информационная система, а также детальный алгоритм ее функционирования (см. раздел 6).

Рассмотренная выше модель регламентирует иерархию и взаимосвязь всей системы требований (уровней). Каждый ее элемент определяется на различных уровнях интерпретации для заинтересованных сторон. Таким образом, взаимосвязи между требованиями, обозначенные на верхнем уровне, однозначно должны определяться уровнем ниже. Это главное правило модели.

Далее рассмотрим примеры использования описанной модели.

3. Применение модели

3.1 Назначение, цели и задачи

Первый этап применения модели — это формализация бизнес-логики будущей информационной системы. Главной задачей на данном этапе является определение назначения и целей создания ИС (рис. 6).



Рис. 6. Модель описания концепции и архитектуры ИС

Для определения целей и задач, особенно в том случае, когда система проектируется для российского заказчика, рекомендуется следовать требованиям ГОСТа 34. В рассматриваемом случае актуальными являются следующие требования:

- назначение системы должно определять вид автоматизируемой деятельности (управление, проектирование и т. п.) и перечень объектов автоматизации, на которых предполагается ее использовать;

- цели создания системы должны регламентировать наименования и требуемые значения технических, технологических, производственно-экономических или других показателей объекта автоматизации, которые должны быть достигнуты в результате создания автоматизированной системы.

При описании целей следует использовать такие языковые конструкции, в которых будут присутствовать категории изменения: «улучшить», «увеличить», «уменьшить», «повысить», «снизить», «упорядочить», «предоставить» и т. д. Из контекста читателю должно быть понятно, что разрабатываемая система действительно улучшит жизнь пользователя, автоматизировав его деятельность; должно присутствовать сравнение типа «есть — должно быть». Кроме того, рекомендуется указывать критерии оценки достижения целей системы, т. е. то, каким образом мы оценим выполнение системой поставленных целей.

Еще одним подходом к описанию целей может быть способ формирования цели по технологии SMART (мнемоническая аббревиатура, используемая в менеджменте и проектном управлении для определения целей и постановки задач — от англ. *Specific, Measurable, Achievable, Realistic, Timely*). Метод отличается от описанного выше в первую очередь тем, что позволяет в одном предложении понять не только «что, зачем, кто», но и «где? когда? как». При определении цели данным методом она по умолчанию содержит критерий для оценки достижения.

Кроме того, есть еще один полезный прием. Он позволяет в ходе интервью заинтере-

сованных сторон правильно сформулировать цели, отталкиваясь от множества сформулированных задач по SMART. На первом этапе интервью необходимо заполнить таблицу со всеми атрибутами. Из полученного множества задач необходимо убрать дубликаты, найти пересечения и взаимоисключения, максимально сократив список. Имея данный список, проведя «проектную сессию», можно сформулировать то назначение и цели системы, которые будут удовлетворять всему объему поставленных задач.

3.2. Модель бизнес-процессов

Далее для получения единого видения (от англ. *vision*) проектируемой информационной системы у всех заинтересованных лиц необходимо формализовать алгоритмы действий, выполняемые разрабатываемой информационной системой. Как известно, алгоритм представляет собой последовательный набор инструкций, определяющий порядок действий для решения задачи за установленное время. Для описания подобного рода алгоритмов целесообразно использовать модели бизнес-процессов.

Существует множество методологий описания бизнес-процессов [9], применяемых в различных ситуациях, однако выбор конкретной методологии в любом случае зависит от контекста решаемой задачи. Основными требованиями к модели являются наглядность последовательности выполнения операций, а также явное описание данных и документов, используемых в подпроцессах. Модель должна полно и однозначно описывать алгоритм действий, происходящих в ИС. Точность построенной модели должна быть такого уровня, который позволит понять логику работы системы и перейти к детальному формированию требований.

С ориентацией на стандарты серии ГОСТ 34 модель бизнес-процессов может быть представлена в виде комплексов задач с соответствующей декомпозицией на функции, выполняемые системой.

На рисунке 7 представлен пример модели бизнес-процессов (см. уровень A.0), в которой выделены комплексы задач (см. уровень A.1 — A.6). Каждый подпроцесс описывается в формате нотации «flowchart» с описанием входных и выходных данных для каждой функции, выполняемой системой.

Когда модель бизнес-процессов построена, можно проследить, как назначение и цели системы на верхнем уровне закрепляют рамки автоматизируемого бизнес-процесса (взаимосвязь цели функционирования и характеристики объекта автоматизации). На данном этапе очевидно, что формализация модели бизнес-процессов является ключевым фактором для принятия множества управленческих решений в ходе согласования и определения рамок проекта по автоматизации объекта, проектирования его архитектуры.

3.3. Варианты использования системы (use cases)

Следующий шаг систематизации требований должен обеспечить общее понимание концепции разрабатываемой информа-

ционной системы между бизнес-аналитиком, архитектором и разработчиком. Этот процесс определяет переход от концептуального уровня на архитектурный (см. раздел 2.2). Для обеспечения данного понимания наилучшим образом подходит описание так называемых вариантов использования (англ. *use cases*), опирающихся на разработанную модель бизнес-процессов.

Практика создания вариантов использования как средств уточнения требований к поведению информационных систем и бизнес-процессов широко используется в мировой практике [10]. Варианты использования обеспечивают комплексное понимание функциональных требований, сопровождение которых необходимо на всех этапах жизненного цикла (англ. *life-cycle*) системы, показывая, как именно будет применяться система в различных ситуациях.

На первый взгляд идея создания вариантов использования кажется довольно простой. Тем не менее, определяя (формализуя) набор вариантов использования, сначала необходимо закрепить уровень их детализации. Для этого поставим вопрос: с какой целью будем описывать данный *use case*, и что получим в результате? Постановка це-



Рис. 7. Описание комплексов задач и модель бизнес-процессов

лей и задач каждого компонента требований позволяет определить требуемый уровень детализации, а также не писать лишнего, что экономит время формализации требований (снижает трудозатраты).

Формализация вариантов использования ни в коем случае не является конечным результатом проектирования ИС. Скорее даже наоборот, это один из первых этапов, обеспечивающий понимание функциональных требований к системе. Обозначить роли акторов (англ. *actor*), сущности и варианты использования — вот то, с чего необходимо начинать определение функционала и пользователей будущей информационной системы.

Варианты использования определяют набор реализуемых и ошибочных последовательностей, которые могут совершаться при взаимодействии актора и системы (подсистемы / класса) для достижения некоторой цели. Совокупность вариантов использования дает разработчику комплексное понимание для проектирования архитектуры приложения, осуществляемого на фазе разработки (англ. *elaboration*). На этом принципе построена, в частности, методология разработки программного обеспечения *Rational Unified Process* [11], созданная компанией *Rational Software* [12].

Таким образом, по результатам анализа назначения и целей описания вариантов использования можно определить ключевую предпосылку систематизации требований: варианты использования являются связующим звеном между пониманием достижения системой целей с позиций всех заинтересованных сторон, реализуя их бизнес-процессы посредством архитектурных и функциональных требований к информационной системе.

Формирование документа, содержащего разделы 3.1, 3.2, 3.3 уже на этом этапе может быть значимым содержанием концепции создания информационной системы (англ. *vision*). Данный документ — отправная точка для закрепления единого видения у всех заинтересованных лиц к автоматизируемым

процессам. Следовательно, на описываемом этапе были представлены общие подходы к определению данных разделов, далее рассмотрим принципы формализации и систематизации требований.

4. Взаимосвязь, классификация и кодирование требований

4.1. Классификация требований

Ранее описывалось, что модель систематизации определяет комплексную взаимосвязь бизнес-процессов, вариантов использования и групп технических требований. Однако набор групп требований должен находиться в каждом случае отдельно, отталкиваясь от эталонной модели (например, по документу *software requirements specification* [13], или составу *types requirements* [8]). Для конкретной информационной системы в ходе ее проектирования необходимо обозначить свой, уникальный набор групп, целостно описывающий требования заинтересованных лиц.

Для того чтобы возможно было построить взаимосвязи (ссылки) между требованиями, а также для повышения удобства их использования и восприятия, необходимо определить коды для всех групп описаний¹ информационной системы (англ. *description*). Для удобства результат классификации рекомендуется помещать в начале документа, что позволяет сразу установить состав технологического содержания документа. В таблице 1 приведен пример классификации требований.

Фактически представленная структура является основной для содержания структуры документа, так как включает все разделы описания информационной системы. Единственное отличие конечного вида — то, что каждое из этих описаний отобразится

¹ Термин «описание» используется, чтобы показать, что описание бизнес-процессов, вариантов использования и классов и характеристик пользователей не является требованиями к информационной системе, в классических терминах ГОСТ 34.

Таблица 1

Классификатор элементов описания

Код	Элемент описания	Раздел
BP	Модель бизнес-процессов (описание комплексов задач)	Описание бизнес-логики
U	Описание классов и характеристик пользователей	
V	Описание вариантов использования	
FA	Требования к функциям (задачам), выполняемым системой	Основные требования к системе
I	Требования к интерфейсу пользователя	
D	Требования к описанию (составу) данных	
R	Требования к отчетам	
AR	Требования к правам доступа	
A	Требования к администрированию, управлению доступом и безопасностью системы	
P	Требования к средствам интеграции	
F	Общие функциональные требования	Дополнительные требования к системе
C	Требования к справочникам и классификаторам системы	
IS	Требования к безопасности	
RD	Требования к надежности	
T	Требования к тестированию	
M	Требования к математическому обеспечению	Требования к видам обеспечения
TS	Требования к техническому обеспечению	
SR	Требования к программному обеспечению	

на всех уровнях представлений в модели (см. раздел 2.2).

4.2. Взаимосвязь требований

На концептуальном уровне взаимосвязь между видами требований описывает отношения между их элементами согласно модели «сущность-связь» (англ. *entity-relationship model*). Отношения между элементами могут иметь как прямую, так и опосредованную связь. Прямая связь представляет собой отношения между элементами групп требований: «один к одному», «один ко многим».

Для лучшего понимания рассмотрим несколько примеров для прямых связей:

- вариант использования «Расчета затрат на общехозяйственные расходы» по принципу «один к одному» относится к алгоритму работы функции «Расчет затрат на общехозяйственные нужды»;

- вариант использования «Расчета затрат на общехозяйственные расходы» по принципу «один ко многим» относится к отчетам «Административно-управленческие расходы» и «Амортизационные отчисления».

К опосредованной связи относится вид связи «многие ко многим» (ее невозможно

отобразить, однако проблему решает преобразование в две связи «один ко многим»). Приведем пример такой связи: требования к отчетам по принципу «многие ко многим» относятся к требованиям к описанию данных. Но эта опосредованная связь прослеживается прямыми связями «один ко многим» через различные варианты использования.

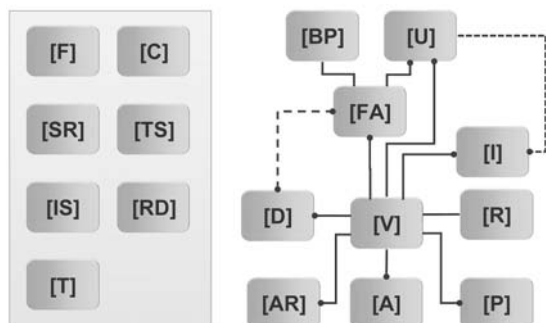


Рис. 8. Взаимосвязь требований

Рассмотрим пример взаимосвязи основных требований (рис. 8). Центральным элементом требований являются варианты использования [V]. Описание каждого из элементов (определенный вариант использования) ссылается:

- на описание классов и характеристик пользователей [U] (связь типа «один к одному», так как каждый вариант использования определяется для конкретного пользователя);
- требования к функциям (задачам), выполняемым системой [FA];
- требования к отчетам [R];
- требования к интерфейсам пользователей [I];
- требования к администрированию, управлению доступом и безопасностью системы [A];
- требования к правам доступа [AR];
- требования к описанию (составу) данных [D].

Функции (задачи), выполняемые системой [FA] в свою очередь по принципу «один к одному» ссылаются на описания бизнес-процессов [BP] и на требования к средствам

интеграции [P]. Остальные (дополнительные) требования в данном случае не имеют непосредственной прямой связи между требованиями и описываются в следующем составе:

- общие функциональные требования [F];
- требования к справочникам и классификаторам [C];
- требования к безопасности [IS];
- требования к надежности [RD];
- требования к тестированию [T];
- требования к математическому обеспечению [M];
- требования к техническому обеспечению [TS];
- требования к программному обеспечению [SR].

Некоторые из требований рекомендуется определять прямой связью. Например, требования к тестированию системы должны быть однозначно «завязаны» на варианты использования или требования к алгоритмам работы функций (в зависимости от подходов к тестированию).

Прямые взаимосвязи между различными элементами-требованиями в документе рекомендуется определять перекрестными ссылками. Чтобы понимать, на какой конкретный элемент мы ссылаемся, необходимо обозначить подход к кодированию элементов.

4.3. Кодирование требований

В разделе 4.1 был определен подход к кодировке требований (если быть точным, в данном случае это виды требований). Каждому виду (группе) требований соответствует некий первичный код, используемый в качестве первичного элемента в общей кодировке.

Далее подход к кодированию требований определяется иерархией элементов (рис. 9):

- Группировка требований [«КОД».XX]
 - Требование [«КОД».XX.XX]
- Спецификация [«КОД».XX.XX.XX]

Каждая группа требований может содержать множество элементов-требований, которые по смыслу рекомендуется объединить



Рис. 9. Кодирование требований

в группу. Таким образом, группировка представляет собой смысловое объединение различных требований. Рассмотрим пример: в группу требований «Расчет затрат» могут входить элементы «Расчет затрат на хозяйственные расходы», «Расчет затрат на общепроизводственные расходы» и др.

Следующий элемент кода — это номер элемента из множества требований в рамках группы, т.е. собственно сами требования. Далее к каждому требованию может прилагаться его спецификация (последний элемент), в которой содержится техническая информация для аналитиков и разработчиков. Состав спецификации зависит от вида требований (пример спецификации рассмотрен в разделе 6).

5. Описание групп требований

5.1. Основные группы

В разделах 3 и 4 было определено отличие описания бизнес-логики от требований к информационной системе. Однако при декомпозиции от комплексов задач до конкретных функций, выполняемых системой, нельзя обойтись без их привязки к вариантам использования и классам пользователей. Поэтому на уровне описания системного аналитика необходимо формализовать уже классифицированные ранее элементы:

- [V] Описание вариантов использования.

- [U] Описание классов и характеристик пользователей.

Кроме того, в рассматриваемом примере к основным требованиям к системе относятся следующие группы (рис. 6, табл. 1):

- [F] Общие функциональные требования.
- [FA] Требования к функциям (задачам), выполняемым системой.
- [I] Требования к интерфейсу пользователя.
- [D] Требования к описанию данных.
- [T] Требования к тестированию.

Описание групп и элементов требований (в рамках вида) рекомендуется отображать в форме таблицы, чтобы удобнее ориентироваться по всей структуре этого вида требований. Эталонная шапка таблицы — три столбца «Код», «Наименование требования» и «Примечание». Однако в зависимости от вида требований таблица может дополняться другими необходимыми (в зависимости от контента) столбцами. Далее подробнее рассмотрим данные виды требований.

5.2. [V] Описание вариантов использования

В настоящем разделе описывается состав вариантов использования информационной системы. Как уже рассматривалось ранее, эта группа является ключевым связующим звеном для всех требований к системе. В таблице 2 представлен пример оформления вариантов использования, в случае если один вариант использования определяет одну роль пользователя. Далее в разделе 6 приведен пример детализации описания варианта использования.

5.3. [U] Описание классов и характеристик пользователей

Описание классов и характеристик пользователей может содержать следующие разделы:

- Классы пользователей.
- Общее описание задач пользователей.
- Требования к правам доступа.

Таблица 2

Требования к вариантам использования

Код	Вариант использования	Пользователь
V.01.00	Требования к основным вариантам использования	
V.01.01	AS — Варианты использования для роли Администратор	AS
V.01.02	AN — Варианты использования для роли Аналитик	AN
V.01.03	RU — Варианты использования для роли Руководитель	RU
V.01.04	OP — Варианты использования для роли Оператор	OP
V.01.05	PO — Варианты использования для роли Учреждение ПО	PO
V.01.06	NP — Варианты использования для роли Незарегистрированный пользователь	NP

В этой группе требований следует описывать характеристики пользователей, выполняемые ими задачи и их права доступа. Для удобства стоит обозначить соответствующим кодом каждый класс пользователя (т.е. его роль). В таблице 3 представлен пример описания классов (ролей) пользова-

телей. На следующем этапе (описание задач и прав доступа) можно детализировать требования на основе общего подхода к систематизации требований, только с привязкой к коду конкретного пользователя.

Далее перейдем к разделу основных требований к информационной системе.

5.4. [F] Общие функциональные требования

Общие функциональные требования могут содержать следующие элементы описания:

- Общие требования.
- Формирование информации.
- Представление информации.

В настоящем разделе преимущественно содержатся требования, применимые к другим элементам требований или к системе в целом. Например, система должна хранить историю изменения значений или иметь возможность изменения настроек профиля пользователя. В таблице 4 представлен пример общих требований.

5.5. [FA] Требования к функциям, выполняемым системой

Требования к функциям, выполняемым системой, могут содержать следующие разделы:

- Описание алгоритмов работы функций.

Таблица 3

Классы и характеристики пользователей

Роль	Код	Описание
Администратор	AD	Лицо, отвечающее за обеспечение целостного функционирования системы. Администратор обладает максимальными правами
Аналитик	AN	Лицо, отвечающее за содержательное функционирование системы. Строит рейтинги учреждений, создает проект премирования на основе рейтинга
Руководитель	RU	Получает агрегированную информацию по формированию рейтингов учреждений и распределению премий
Оператор	OP	Лицо, выполняющие работы по информационному наполнению системы и контролю корректности данных и т.п.
Учреждение	FI	Филиал организации. Имеет доступ к своей персональной информации

Таблица 4

Общие функциональные требования

Код	Требования	Примечания
F.01.00	Общие требования	
F.01.01	Работа пользователя с системой должна быть организована в режиме он-лайн через тонкий клиент (интернет-браузер)	С использованием одного из браузеров: IE версии 7 и выше, Firefox 3.6 и выше, Chrome 10 и выше, Safari 5 и выше
F.01.02	В системе должен быть предусмотрен пользовательский интерфейс для редактирования логической структуры портала и публикации различных видов информационных материалов	Виды информационных материалов: Новостные сообщения, статьи, документы формата MS Word
F.01.03	Система должна обеспечивать доступ к информационным материалам посредством интернет-портала, поддерживающего навигацию пользователей в соответствии с многоуровневым (иерархическим) классификатором	

• Требования к качеству реализации каждой функции.

• Временной регламент реализации каждой функции.

Данная группа содержит требования к функциям (задачам), выполняемым системой. Функции (задачи) есть продолжение (декомпозиция) комплексов задач, определяемых бизнес-процессами. Функции (задачи) также могут вырабатываться на основе вариантов использования, при применении объектного подхода к проектированию информационной системы. Однако первый вариант наиболее удобен для по-

строения процессоориентированных систем. В данном случае примером может являться та же группа требований «Расчет затрат», в которую могут входить элементы «Расчет затрат на общехозяйственные расходы», «Расчет затрат на общепроизводственные расходы» и др. В этом случае в полном соответствии (прямая связь «один к одному») с бизнес-процессами определяются функциональные алгоритмы с аналогичными названиями.

В примере (табл. 5) показана ситуация, когда в табличной форме указывается код функции бизнес-процесса и пользователи,

Таблица 5

Требования к алгоритмам работы функций

Код треб.	Код функции	Функция	Пользователи
FA.01.00	A.1.	Анализ и верификация исходных данных	
FA.01.01	A.1.1.	Загрузка массива данных	AS, OP
FA.01.02	A.1.2.	Верификация данных	AS, OP, RU*
FA.01.03	A.1.3.	Утверждение данных	AS, OP
FA.06.00	A.6.	Формирование государственных заданий	
FA.06.01	A.6.1.	Расчет затрат на оказание образовательной услуги	AS, AN
FA.06.02	A.6.2.	Расчет затрат на общехозяйственные нужды	AS, AN
FA.06.03	A.6.3.	Расчет затрат на содержание имущества	AS, AN
FA.06.04	A.6.4.	Формирование проекта	RU, AS, AN

Таблица 6

Требования к интерфейсу пользователя

Код	Требования	Примечания
I.01.00	Общие требования	
I.01.01	Пользователь должен иметь возможность доступа к информации путем перехода по гиперссылкам системы	навигационный
I.01.02	Пользователь должен иметь возможность доступа к информации	поисковый
I.01.03	Не должно быть кнопок без имени или не помеченных специальной иконкой	
...
I.02.00	Требования к элементам управления	
I.02.01	Предупреждение об обязательности заполнения поля	
I.02.02	Подчеркнутый текст — признак того, что элемент становится активным при открытии формы	
I.02.03	Строка заголовка, данные, итоговая строка — недоступны для редактирования, строку данных можно выбирать	

А. В. Симкин

использующие данную функцию и соответствующий бизнес-процесс.

Таблица 7

Требования к описанию данных

5.6. [I] Требования к интерфейсу пользователя

Требования к интерфейсам пользователя могут содержать следующие разделы:

- Описание разделов системы.
- Макеты экранных форм.
- Алгоритмы взаимодействия.

Требования к интерфейсам зачастую представляют собой три смысловых блока: общие требования к интерфейсам, требования к элементам управления и детализируемые требования к конкретным интерфейсам пользователя (экранным формам). Первые два зачастую описываются текстовой информацией, последнюю группу требований для лучшего понимания рекомендуется визуализировать в виде макета интерфейса. В примере (табл. 6) представлены требования к интерфейсу пользователя.

5.7. [D] Требования к описанию данных

Требования к описанию данных могут содержать следующие элементы:

Код	Требования	Примечания
D.01.00	Требования к составу метаданных объекта данных	
D.02.00	Требования к составу данных	
D.02.01	Нормативные затраты	FA.06.01
D.02.02	Налоги	FA.06.02
D.02.03	Тарифы	FA.06.03
D.02.04	Затраты учреждения на тепловую энергию в отчетном периоде	FA.06.03
D.02.05	Затраты учреждения на электроэнергию в отчетном периоде	FA.06.03
D.03.00	Требования к представлению данных	
D.03.01	A.1.2. Верификация данных	FA.01.02
D.03.02	A.3.8. Расчет агрегированных показателей	FA.03.08

- Описание метаданных.
- Описание состава данных.
- Описание представлений данных.

В настоящем разделе описываются требования, содержащие информацию о данных, хранящихся и обрабатываемых в информационной системе. Описание метаданных представляет собой информацию об атрибутах и их характеристиках. Описание состава данных включает необходимые табличные формы и перечень соответствующих атрибутов. Описание представлений данных содержит информацию о виде табличных форм и отчетов. Таким образом, определяются требования к описанию и составу данных (от их атрибутов до визуального представления отчетных форм). В таблице 7 представлен пример описания требований к данным.

5.8. [Т] Требования к тестированию

Требования к тестированию могут содержать следующие элементы:

- Описания типов тестов.
- Программа и методика испытаний.
- Шаблон протокола тестирования.

В этом разделе следует определить и описать подходы к тестированию систе-

мы и приемке системы заказчиком (заинтересованными лицами). Проведение тестов может быть привязано как к конкретным алгоритмам работы функций (компонентное тестирование), так и к вариантам использования (функциональное тестирование), или, например, к интерфейсам пользователя (юзабилити-тестирование и тестирование интерфейса пользователя), а также к другим видам технических требований. Для этого устанавливается прямая связь (вида «один к одному» или «один ко многим») и описываются требования к тестированию каждой отдельной функции (в случае с функциональным тестированием).

Для формальных процедур тестирования рекомендуется определять формат протокола тестирования в целях сокращения в будущем трудозатрат на оформление результатов тестирования. Подобные протоколы должны подписываться участками приемочной комиссии при демонстрации системы или ее сдачи (показа) заказчику.

6. Спецификации требований

Спецификации требований предназначены для описания конкретных реализаций

Таблица 8

Описание варианта использования

Раздел Use-case	Содержание
Класс пользователя	ОР
Описание	Лицо, выполняющее работы по информационному наполнению, контролю корректности данных. Осуществляет процесс загрузки и верификации данных
Нормальное направление	[V. 01.01.01] Загрузка данных Условие: наличие файлов для загрузки 1. Пользователь переходит в один из разделов меню «Исходные данные» 2. Пользователь выбирает вкладку «Загрузка данных» (см. Требование [U.04.01.01] — Интерфейс «Исходные данные — загрузка») 3. Пользователь выбирает шаблоны Excel для загрузки и нажимает кнопку «Загрузить»
Обработка исключений	Проверка соответствия формата шаблона загрузки Excel, в случае наличия отличий отказ обработки, сообщение пользователю
Специальные требования	Механизм импорта из Excel

функций, вариантов использования, интерфейсов и т. п. Именно в них описывается сам элемент группы технических требований и его особенности. Например, если это алгоритм функции, то описываются все шаги его реализации, особенности, исключения и т. п., т. е. дается вся информация, которая понадобится разработчику при реализации этой функции в виде программного кода. В таблице 8 приведен вариант описания спецификации для варианта использования.

Заключение

В настоящей статье рассмотрен не только подход к систематизации, отражающий комплексное применение различных методологий при проектировании информационных систем, но и правила, и практика его применения в виде рамочной модели систематизации требований.

Несмотря на ограниченный состав рассмотренных методов систематизации требований, следует отметить, что конечный состав модели систематизации должен определяться исходя из специфики решаемой задачи, т. е. разработанная и представленная модель может быть дополнена или сокращена.

Таким образом, сформировав технические требования к ИС с подобным уровнем детализации, можно обеспечить сокращение трудозатрат на формирование проектной документации, определяемой требованиями ГОСТ 34. Например, из представленного выше перечня требований можно получить следующие документы:

- техническое задание по ГОСТ 19 и ГОСТ 34;
- схему функциональной структуры;
- пояснительную записку;
- описание постановки задач (комплекса задач);
- описание информационного обеспечения системы;
- программу и методику испытаний;
- спецификации (для программиста).

Список литературы

1. *George A. Miller*. The Magical Number Seven, Plus or Minus Two // *The Psychological Review*. 1956. Vol. 63. P. 81–97.
2. *Берталанфи Л. фон* Общая теория систем — Критический обзор // *Исследования по общей теории систем*. М.: Прогресс, 1969. С. 23–82.
3. Системный анализ // Википедия. Открытая энциклопедия. URL: http://ru.wikipedia.org/Системный_анализ (дата обращения: 20.09.2012).
4. *Качала В. В.* Основы теории систем и системного анализа: учеб. пособие для вузов. М.: Горячая линия — Телеком, 2007. — 216 с.
5. *Данилин А., Слюсаренко А.* Архитектура и стратегия. Инь и Янь информационных технологий предприятия. М.: Интернет-Ун-т Информ. Технологий. 2005. — 504 с.
6. *John A. Zachman*. The Zachman Framework™: The Official Concise Definition // *Zachman International Web Site*. URL: <http://zachmaninternational.com/index.php/home-article/13> (дата обращения: 20.09.2012).
7. ISO/IEC/IEEE 42010 Systems and software engineering — Architecture description // JTC 1/SC 7. ISO publications, 2011.
8. *Systems Engineering Fundamentals* // Department Of DefenseSystemManagementCollege. Fort Belvoir, Virginia, 2001.
9. *Ковалев С. М., Ковалев С. В.* Современные методологии описания бизнес-процессов — просто о сложном // *Консультант директора*. 2004. № 12.
10. *Коберн А.* Современные методы описания функциональных требований к системам. М.: Лори, 2002. — 266 с.
11. RationalUnifiedProcess // Википедия. Открытая энциклопедия. URL: http://ru.wikipedia.org/wiki/Rational_Unified_Process (дата обращения: 01.02.2013).
12. RationalSoftware // Википедия. Открытая энциклопедия. URL: http://ru.wikipedia.org/wiki/Rational_Software (дата обращения: 01.02.2013).
13. *Stellman A., Greene J.* Applied software project management // O'Reilly Media, 2005. — 308 p.