Подготовка функциональных спецификаций для разработки корпоративных информационных систем на примере пользовательских ABAP-программ в SAP ERP (часть 2)

Степанов Дмитрий Юрьевич

Аннотация: в работе рассматривается процедура Fit/Gap-анализа и последующее оценивание разработок, описывается RICEF-классификация и принципы проектирования программ, дается структура и содержание функциональной спецификации на разработку на примере системы SAP ERP, вводятся типовые функциональные требования для отражения в документе спецификации.

4. Типовая структура спецификации на разработку

Завершив подготовительное концептуальное проектирование программы и идентифицировав стандартные таблицы по хранению данных в ERP-системе, приступаем к написанию функциональной спецификации. Содержание спецификации зависит от категории RICEF-разработки, однако можно выделить следующие типовые разделы документа:

- описание исходных бизнес-требований;
- модель TO-BE;
- допущения;
- концепция решения;
- логика взаимосвязи экранов, их полей и алгоритмов заполнения;
- сценарии функционально-модульного тестирования;
- роли и полномочия.

Бизнес-требования, описание процесса с учетом текущей разработки в модели TO-BE, а также концепция решения необходимы для того, чтобы пользователи могли однозначно идентифицировать исходные потребности в текущей разработке и ознакомиться с верхнеуровневым решением, не вдаваясь в технические подробности. Последнее весьма существенно, так как одним из подтверждающих документа спецификации всегда является бизнес-представитель, далекий от таблиц баз данных, языков программирования и программной архитектуры.

Раздел допущений введен неслучайно. Общий подход к обработке требований заключается в том, что все открытые вопросы и неоднозначности должны быть закрыты и максимально детально уточнены бизнес-пользователями или внешней организацией. К сожалению, это не всегда достижимо, что не отменяет исходные сроки подготовки спецификации. В подобных случаях используются допущения.

Определение 10. *Допущение* – это предположение или гипотеза, выдвижение которых равнозначно самостоятельному ответу на открытый вопрос, что позволяет снизить неопределенность в процессе имплементации КИС.

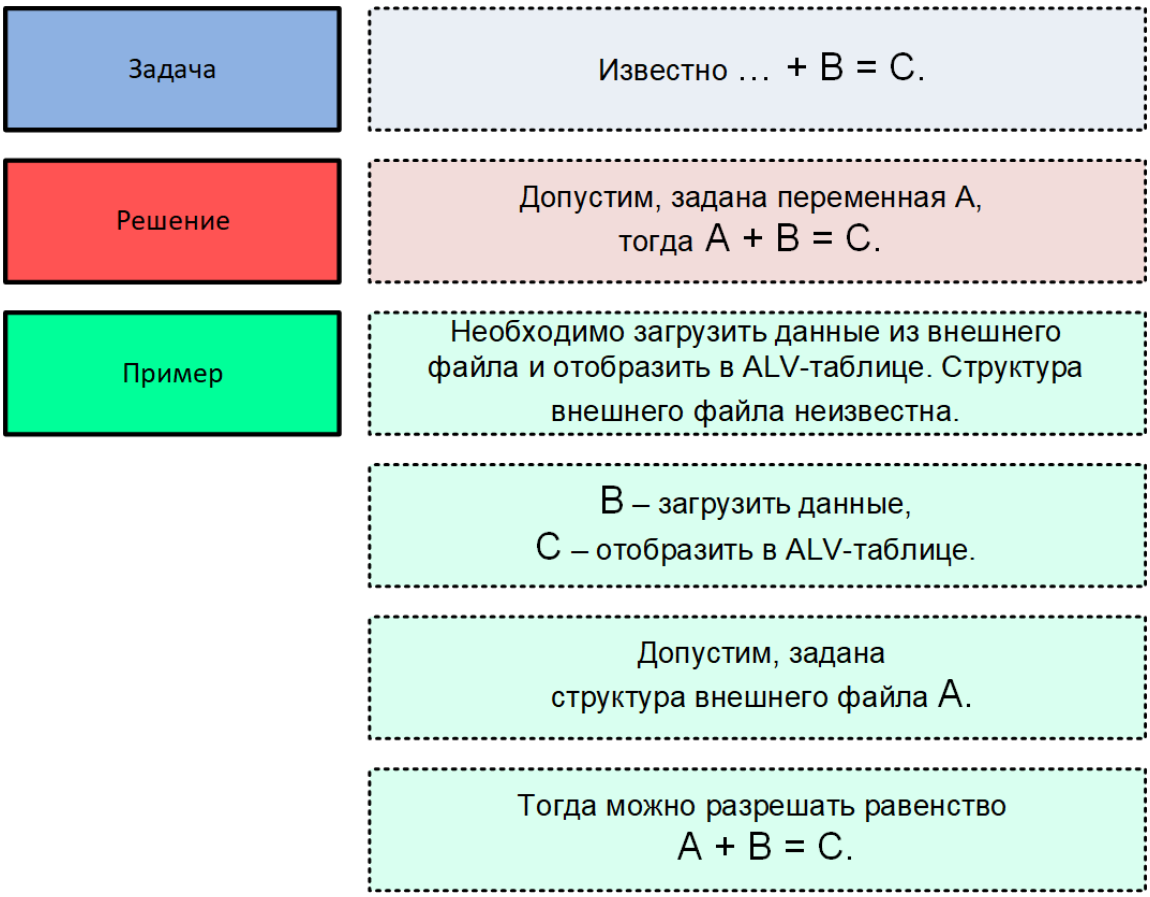


Рис. 23. *Пример использования допущений для исключения неопределённостей*

Следуя изначально продуманной концепции решения, описываются экраны пользовательской программы и логика перехода между ними, кроме того поля, кнопки, функции и подэкраны каждого экрана, а также алгоритмы обработки данных, что обсуждалось ранее в п.3.2-3.3. Детализация указанных элементов – самая трудоемкая задача процесса подготовки спецификации, требующая львиную долю плановых трудозатрат.

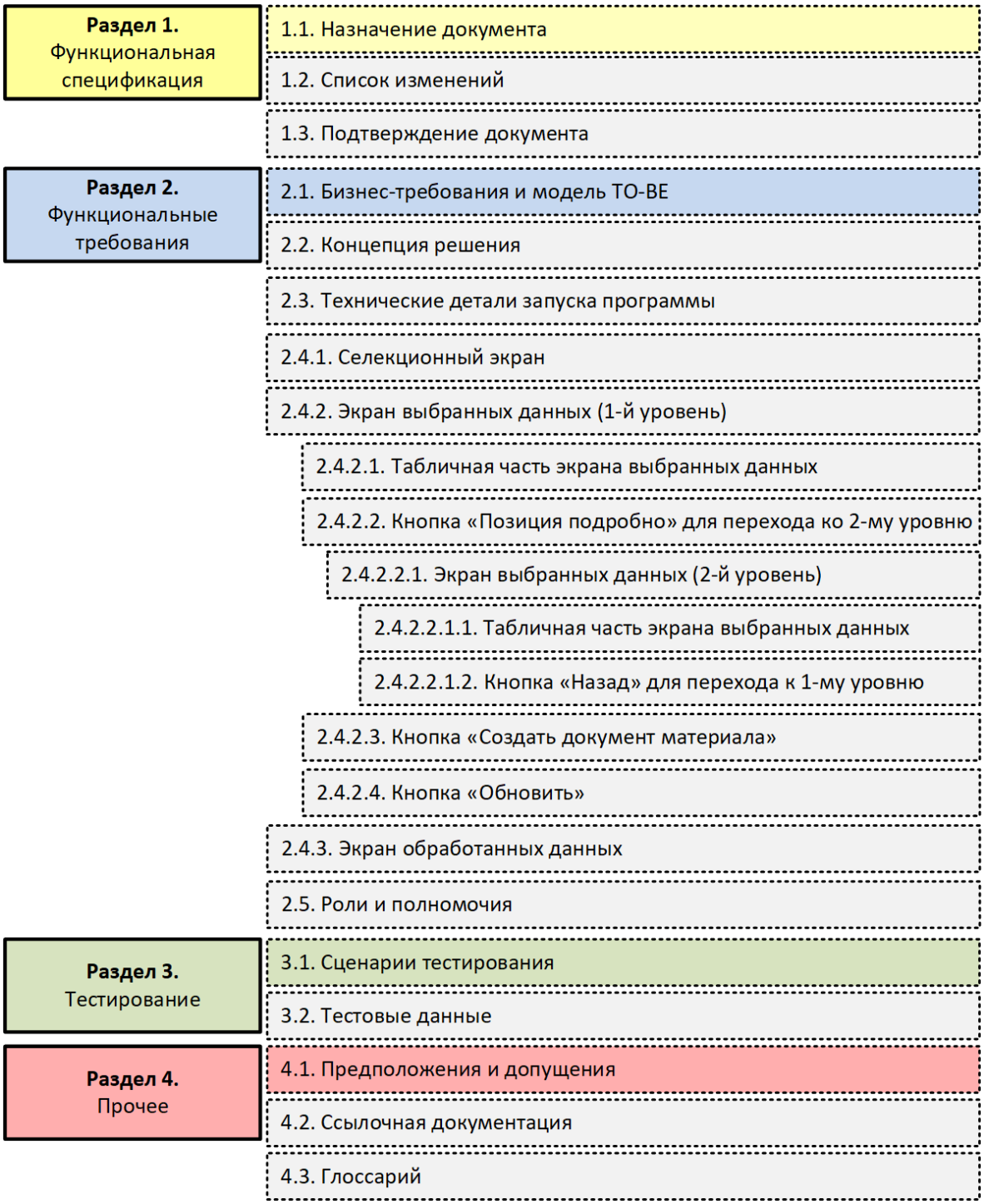


Рис. 24. *Пример структуры спецификации для разработки С-программы по обработке данных*

Проведение испытания разработанной на основе спецификации программы осуществляется в среде разработки путем функционально-модульного тестирования. Сценарии тестирования, дополненные тестовыми данными, приводятся в

одноименном разделе. Документирование и отслеживание последующих видов испытаний: системного, интеграционного и непрерывного тестирований, ведется отдельно, преимущественно в программной системе ALM (Application Lifecycle Management) от компании HP.

Реализованные с нуля отчеты, программы, формуляры и интерфейсы должны быть добавлены в технические роли для возможности использования пользователями в ERP-системе. Присвоение пары «программа – роль» описывается в разделе ролей и полномочий. На рисунке ниже даны несколько примеров содержания спецификации на разработку для получения более полной картины возможных способов наполнения документа.

5. Особенности проектирования программ и их отражение в спецификации на примере SAP ERP

Типовая структура спецификации применяется для моделирования и последующей реализации любой RICEF-разработки. Введенные принципы проектирования программ так же находят свое отражение в документе спецификации. Уточним их до уровня функциональных требований, используя следующую шкалу приоритетов: высокий, средний и низкий. Если принципу соответствует высокий приоритет, значит его несоблюдение не позволит перенести программную разработку в продуктивную ERP-систему, средний – следование принципу обеспечивает унификацию и генерализацию решения, его игнорирование в будущем приведет к ошибке, что потребует доработки программы, и, наконец, низкий приоритет носит рекомендательный характер, относящийся к категории «бантики» (или «Nice to Have» в англоязычной литературе). Результаты детализации принципов до функциональных требований и эмпирическая оценка их приоритетов рассмотрена в пунктах ниже.

5.1. Проектирование всех видов RICEF-разработок

Выделим функциональные требования, относящиеся ко всем видам RICEF-разработок (табл.1). Высокоприоритетными требованиями являются: использование утвержденных правил наименования объектов разработки, а также вынесение константных переменных в отдельный модуль. Первое требование определяет технический регламент наименования переменных, функций, подпрограмм, объектов, структур и прочих составляющих разрабатываемого приложения (Naming Convention, соглашение по наименованию); второе задает порядок ведения константных переменных для программных разработок, как известно, запрещено их явное указание в программном коде. Способы обработки константных значений приведены на рис.25 в привязке к системе SAP ERP.

Таблица 1. *Функциональные требования, применимые ко всем видам разработок*

Вид разработки	Принцип	Функциональное требование	Приоритет
Все разработки RICEF	Развития	Применение утвержденных правил наименования технических объектов	Высокий
Все разработки RICEF	Развития	Вынесение константных переменных в отдельный модуль	Высокий
Все разработки RICEF	Развития	Общий вид решения задачи	Средний
Все разработки RICEF	Контур обратной связи	Информирование пользователя о результатах работы программы	Средний
Все разработки RICEF	Модульности	Разработка и использование функций для часто повторяющихся операций	Средний
Все разработки RICEF	Контур обратной связи	Обновление ролей для разработанной программы	Средний
Все разработки RICEF	Развития	Корректное наименование программных разработок (лингвистическое обеспечение)	Низкий

Функциональные требования, имеющие средней приоритет: общий вид решения задачи, информирование пользователей о результатах работы программы и применение функциональных модулей для часто повторяющихся операций, уже обсуждались нами при рассмотрении принципов развития, контура обратной связи и модульности соответственно. Помимо указанных требований существует также потребность в расширении технических ролей для новых разработанных программ. Действительно, если была реализована дополнительная транзакция в терминах SAP ERP, ее необходимо включить в PFCG-роль пользователя, в противном случае никто не сможет запустить программу в продуктивной КИС.

Обеспечение корректного бизнес-названия программ, отображаемых в пользовательском меню ERP-системы, задает единственное функциональное требование с низким приоритетом. Содержание требования состоит в следующем: при разработке большого числа программ их краткое описание должно ограничиваться определенными правилами с целью унификации и удобства пользования. Например, была разработана программа печати формуляра для приходного ордера (М-4), существует огромное число вариантов наименования разработки: «Программа для

печати формы М-4», «Печать приходного ордера», «М4» и т.д. Схожая задача лингвистического обеспечения возникает в SAP ERP при заведении основной записи материала (Material Master, ОЗМ).

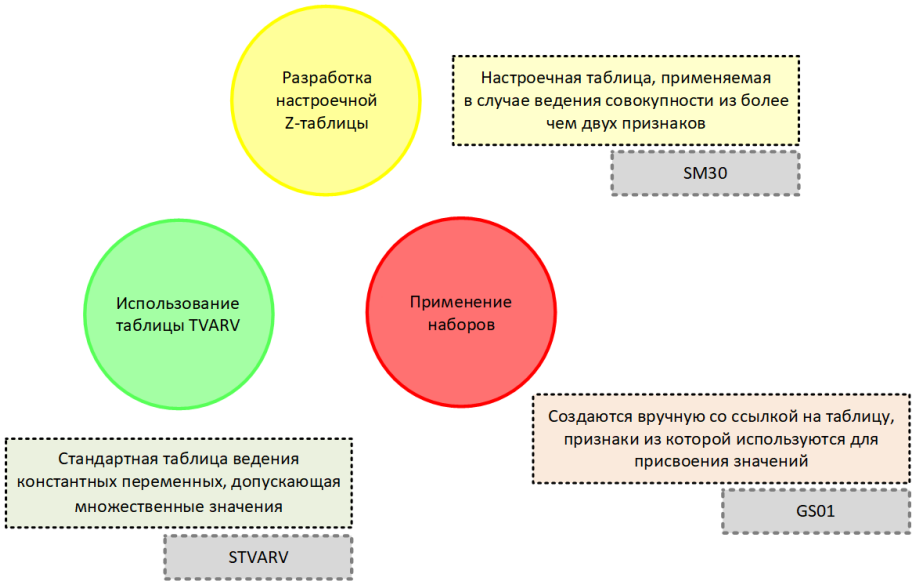


Рис. 25. *Способы ведения константных переменных в SAP ERP с указанием кодов транзакций*

5.2. Особенности проектирования отчетов (R-тип)

Рассмотрим высокоприоритетные функциональные требования для проектирования программных разработок вида отчет: указание организационных уровней и ограничение выбора данных на основе полномочий (табл.2). Эти требования взаимосвязаны, согласно концепции ролей и полномочий пользователь должен видеть в ERP-системе лишь те данные в разрезе оргуровней, на которые ему дано разрешение. Полномочия пользователя могут изменяться, поэтому признак организационного уровня, как то: компания, балансовая единица, завод, склад, складское место и т.д., должен быть вынесен на селекционный экран, что является содержанием первого требования. Суть второго состоит в том, что, если пользователь запросил информацию, но полномочия на просмотр он имеет лишь для части запрошенных данных, КИС должна проинформировать его об этом (рис.26).

Таблица 2. *Функциональные требования, применимые к разработкам вида R-отчет*

Вид разработки	Принцип	Функциональное требование	Приоритет
R-отчет	Развития	Указание организационных уровней на селекционном экране	Высокий

Вид разработки	Принцип	Функциональное требование	Приоритет
R-отчет	Контура обратной связи	Ограничение выбора данных на основе полномочий	Высокий
R-отчет	Функциональности	Отображение формата полей на селекционном экране	Средний
R-отчет	Развития	Вывод максимального количества полей в таблицу отображения данных	Средний
R-отчет	Развития	Исключение сторнированных или помеченных на удаление документов	Средний
R-отчет	Функциональной избирательности	Использование фоновых задач для обработки больших данных	Средний
R-отчет	Функциональной избирательности	Индексация для обработки больших данных в диалоговом режиме	Средний

Далее перейдем к функциональным требованиям, имеющим средний приоритет. Проанализируем требования, задающие выборку данных: отображение формата полей на селекционном экране, вывод максимального числа полей данных по результатам селекции и исключение сторнированных/удаленных документов. Уже говорилось о том, что ERP-системы преимущественно выдают отчеты в табличном виде. Независимо от того, какой набор полей изначально запрашивается для отображения в отчете пользователем, всегда остаются не озвученные, но потенциально необходимые данные. Например, если речь идет о суммах, то вероятнее всего также потребуется вывод кода валюты, количестве – единицы измерения, продукции – описание материала и прочее. Во избежание дальнейших доработок можно заранее заполнить большее число атрибутов отчета нежели заявлено клиентом.

Порядок следования полей, признаки суммирования и фильтрации в отчетах SAP ERP можно кастомизировать, применяя функционал формата полей. Создав единожды необходимый формат, можно запрограммировать его выбор на селекционном экране программы. Как результат, отображение полей в отчете будет единообразно для всех пользователей, однако при необходимости его легко поменять. И, наконец, при поиске данных не следует забывать о том, что отмененные или помеченные на удаление данные малоинформативны для пользователей, поэтому их нужно исключать из отчета или предусматривать соответствующие индикаторы на селекционном экране.

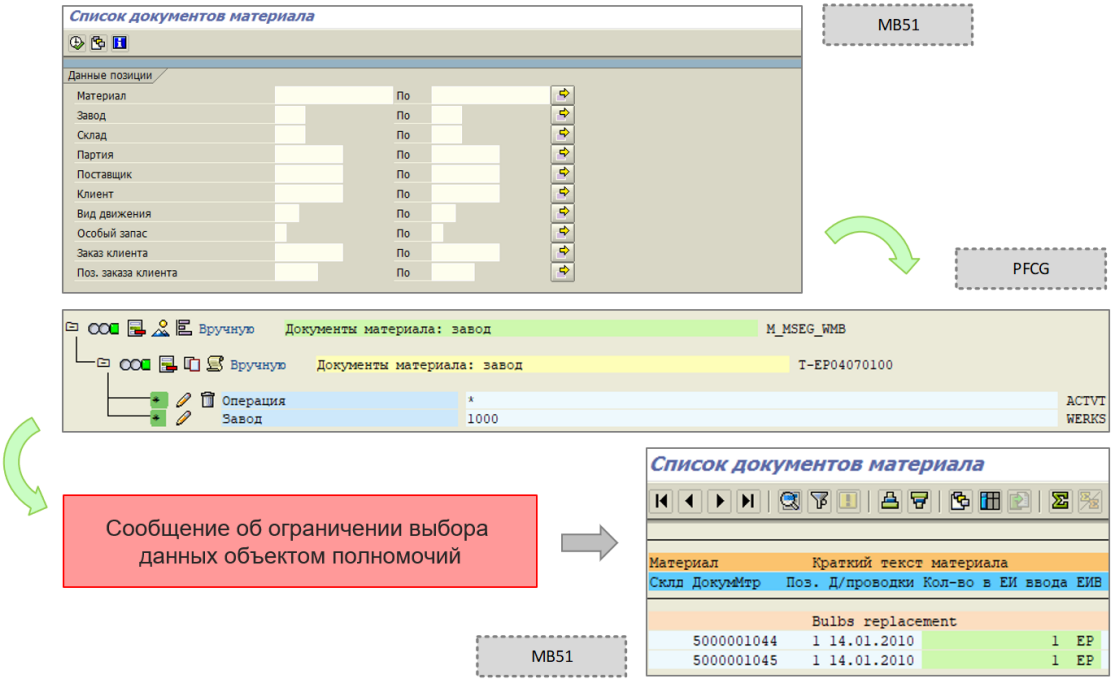


Рис. 26. *Алгоритм ограничения выбора данных согласно PFCG-роли на примере транзакции MB51 в SAP ERP*

Следующие функциональные требования относятся к обработке больших данных: использование фоновых задач и индексация данных в диалоговом режиме (рис.27). На момент подготовки спецификации на разработку технический специалист должен иметь четкое представление о том, для работы с каким объемом данных будет применяться отчет, какова степень детализации предоставляемой информации, а также время отклика. В случае, если отчет должен отражать сводные данные компании, например, за год, то потребуется проектирование фоновых задач, обрабатывающих информацию и сохраняющих их в агрегированном состоянии во временные таблицы SAP ERP. Когда в последующем пользователь запустит отчет, данные будут считываться из заполненных временных таблиц, что обеспечит выигрыш по времени отклика.

Если пользователь запрашивает и ожидает быстро получить не агрегированную информацию из SAP ERP, то возможно применение функционала индексации таблиц баз данных, для чего необходимо выполнить как донастройку системы, так и ее доработку. Начальный запуск отчета будет происходить достаточно долго, что объясняется первичной индексацией данных, однако последующие – на порядок быстрее. Указанные функциональные требования должны быть зафиксированы и отражены в документе спецификации.

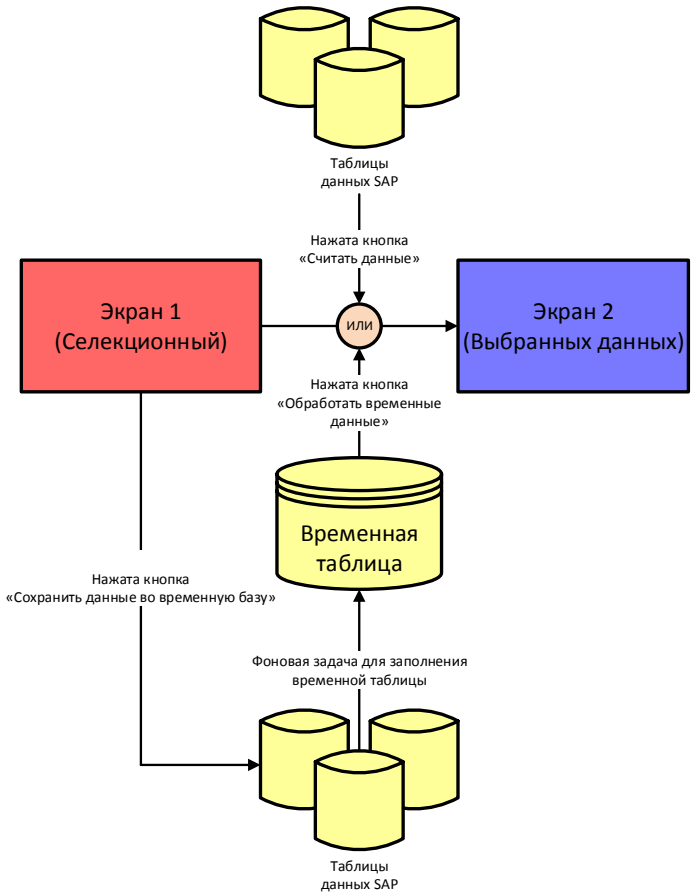


Рис. 27. *Схема обработки больших данных на основе хранения агрегационной информации во временных таблицах SAP ERP*

5.3. Проектирование программ обработки данных (C-тип)

Особенностью программ по обработке информации (тип разработки C) является то, что модифицированные ими стандартные данные ERP должны быть помечены особым образом. Пометку можно организовать двумя способами: путем заведения временной таблицы, в которую заносятся номера обработанных документов, или указанием технической информации на уровне заголовка изменяемого документа. Допустим, вы обновили большое число объектов и потом поняли, что модификацию нужно отменить. Если функциональное требование выше не было реализовано, то придется вручную искать документы для идентификации «ваших».

Таблица 3. *Функциональные требования, применимые к разработкам вида C-программы обработки данных*

Вид разработки	Принцип	Функциональное требование	Приоритет
C-программа обработки данных	Надежности	Пометка обработанных данных	Средний

Вид разработки	Принцип	Функциональное требование	Приоритет
С-программа обработки данных	Надежности	Вывод стандартного описания системных ошибок в экран обработки	Средний
С-программа обработки данных	Надежности	Хранение потока документов при обработке большого числа объектов	Средний

Какова бы не была сложность С-программы, разрабатываемое приложение будет создавать, изменять и удалять данные. Обработка данных потребует организации большого числа технических проверок и отображение соответствующих информационных сообщений пользователю. Нередки случаи, когда текст ошибки от новой разработанной проверки перекрывает стандартное информационное сообщение ERP-системы. Тем самым дублируется и переписывается поведение КИС. Последнее недопустимо, потому что, во-первых, происходит никому не нужное увеличение трудозатрат для подготовки и разработки уже имеющейся логики в базовом комплекте ERP-системы, во-вторых, стандартные ERP-проверки содержат более детализированное описание и зачастую предполагаемые шаги разрешения возникшей ситуации.

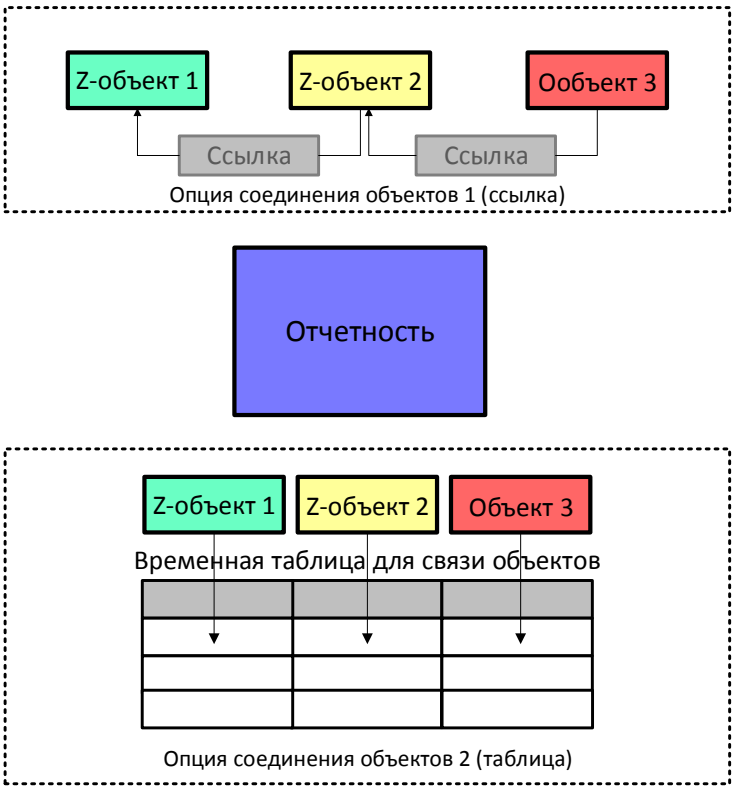


Рис. 28. *Организация потока документов на основе ссылок и временных таблиц баз данных*

Если разрабатываемая программа C-класса порождает новые объекты данных, отсутствующие в КИС-комплекте, рано или поздно возникнет необходимость получения аналитической отчетности для целей мониторинга и контроля. Поэтому создаваемые объекты связывают в поток документов, используя временные таблицы, в которые последовательно записываются номера объектов каждой бизнес-цепочки, или ссылку каждого последующего документа на предыдущий (рис.28). В дальнейшем отчетность строится на базе потока документов.

Рассмотренные функциональные требования: пометка обработанных данных, отображение стандартного описания ошибок и хранение потока документов (табл.3), детализируют принцип надежности, имеют средний приоритет реализации и позволяют существенно сократить число ошибок при разработке и тестировании C-программ в случае их имплементации.

5.4. Моделирование печатных форма (F-тип)

Среди функциональных требований к разработке печатных форм (F-программы) можно выделить выбор данных для заголовка формуляра на основе позиций документа. Часто при печати верхнего колонтитула формуляра данные приходится выбирать не из заголовка, а с уровня позиций документа. В этом случае возникает вопрос, из какой именно позиции, ведь их много? Данное требование постулирует селекцию из первой не удаленной позиции. Оставшиеся требования: «защита от дурака» и применение параметров пользователей, уже обсуждались при рассмотрении принципов неопределенности и «по умолчанию».

Таблица 4. *Функциональные требования, применимые к разработкам вида F-печатная форма*

Вид разработки	Принцип	Функциональное требование	Приоритет
F-печатная форма	Неопределенности	Защита от дурака	Средний
F-печатная форма	«По умолчанию»	Применение параметров пользователя	Низкий
F-печатная форма	Надежности	Выбор данных для заголовка из первой не удаленной позиции документа	Низкий

5.5. Разработка интерфейсов (I-тип)

Начнем рассмотрение функциональных требований, применимых к разработкам вида интерфейс (I-тип), с обзора концептуальной схемы взаимодействия информационных систем (рис.29). Как видно из рисунка, выделяют ИС отправителя и

получателя, а также интеграционную шину (Gateway), служащую преобразователем формата данных, передаваемых между системами. Типичная проблемная ситуация при интеграции ERP-систем: сообщение было отправлено, но не сохранилось в базе данных КИС. Вопрос, в какой момент и где возникла ошибка? Потенциально, каждая из система может быть источником проблемы:

- сообщение не было отправлено системой отправителя;
- до шины не дошло сообщение или она его не перенаправила системе получателя;
- сообщение не пришло в систему получателя или пришло в ином формате, что породило ошибку обработки.

Цель предлагаемых функциональных требований – обеспечение максимальной прозрачности процесса обработки входящих и исходящий сообщений для быстрой отладки непредвиденных ситуаций во избежании проведения детективного анализа (табл.5). Реализация требования по просмотру содержания исходного сообщения, полученного для последующей загрузки в ERP-систему, позволяет выявить случаи не следования установленному формату обмена и заявленной структуре сообщения как на стороне отправителя, так и интеграционной шины.

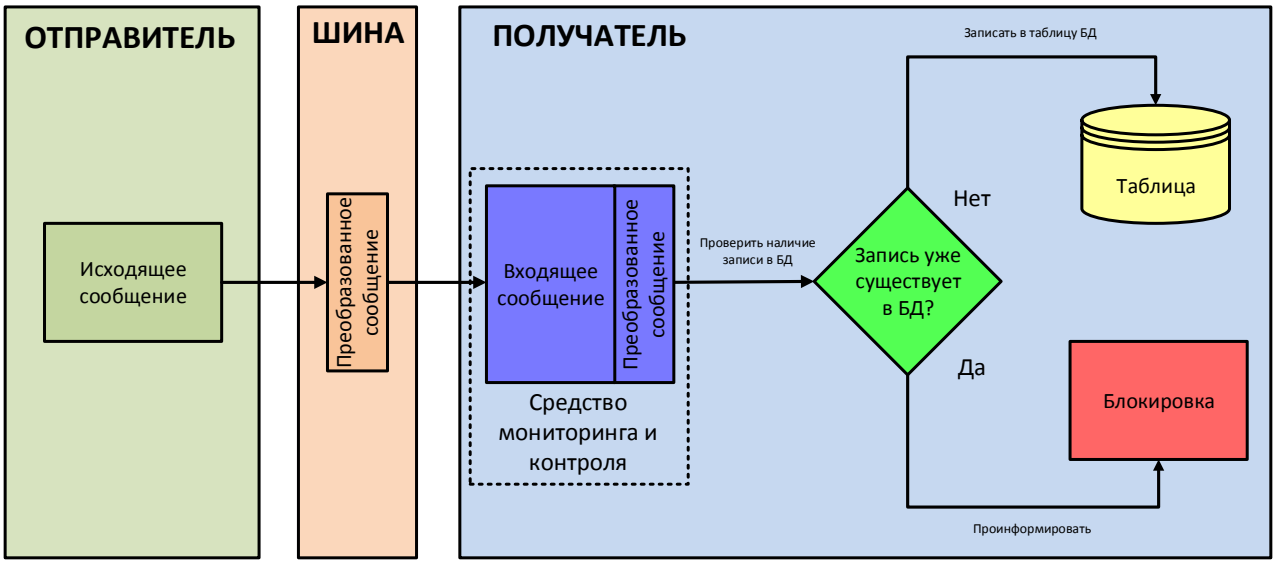


Рис. 29. *Концептуальная схема обмена сообщениями между различными ИС*

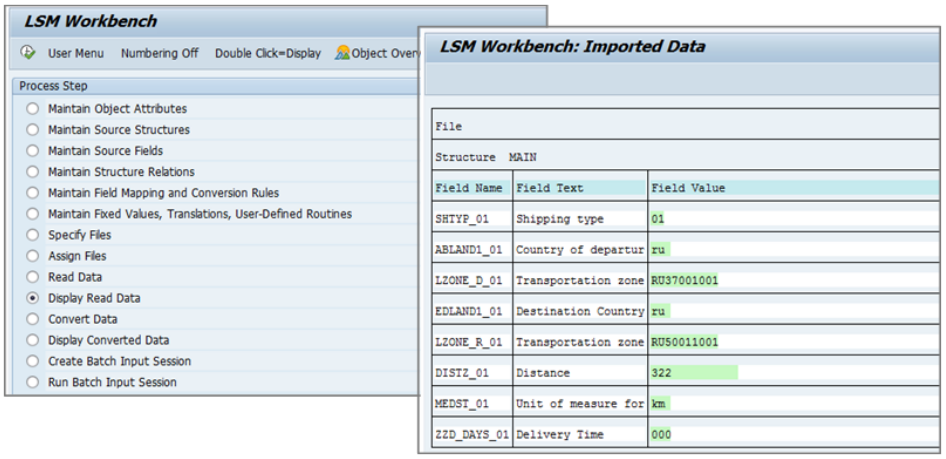
Допустим входящее сообщение успешно получено и имеет согласованные структуру и формат. Реализация механизмов моделирования и просмотра преобразованного сообщения, полученного из исходного для целей получателя, обеспечивает дополнительное средство контроля ошибок (рис.30). Заключительным

функциональным требованием служит блокировка/информирование пользователей в случае, если уже обработанное входящее сообщение пытаются повторно загрузить.

Таблица 5. *Функциональные требования, применимые к разработкам вида I-интерфейс*

Вид разработки	Принцип	Функциональное требование	Приоритет
I-интерфейс	Неопределенности	Возможность просмотра исходного сообщения с данными, полученного для загрузки или подготовленного для отправки	Средний
I-интерфейс	Неопределенности	Обеспечение возможности просмотра преобразованных для загрузки данных	Средний
I-интерфейс	Надежности	Блокировка/информирование повторной загрузки данных	Низкий

а)



б)

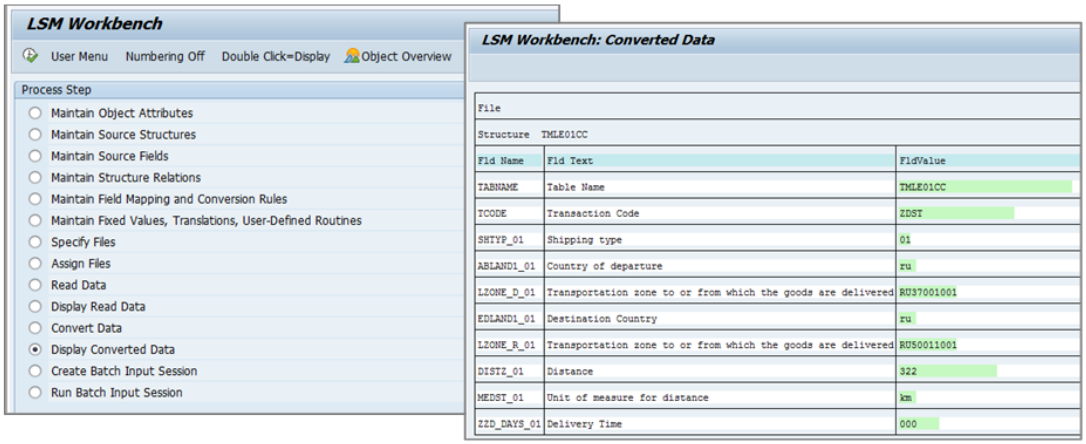


Рис. 30. *Транзакция LSMW системы SAP ERP для просмотра данных к загрузке: а) исходные; б) преобразованные из исходных*

5.6. Работа с расширениями (E-тип)

Среди функциональных требований, релевантных программным расширениям (E-тип), можно выделить следующее – максимальное использование стандартного функционала КИС. Что говорит вот о чем: довольно часто требуется модифицировать приложение и на первый взгляд точка расширения должна быть имплементирована в одну из самых сложных и запутанных программ. Реализация процессов в большинстве ERP-систем строится на основе бизнес-объектов, как-то: заказ, приход, фактура, списание и др., тем самым формируется взаимосвязанный поток документов. Атрибуты создаваемых документов чаще всего определяются автоматически на основе данных из предыдущих объектов. Поэтому, если требуется доработать логику обработки объекта, вполне вероятно, что изменить потребуется алгоритм обработки предшествующих документов, но не текущего (рис.31).

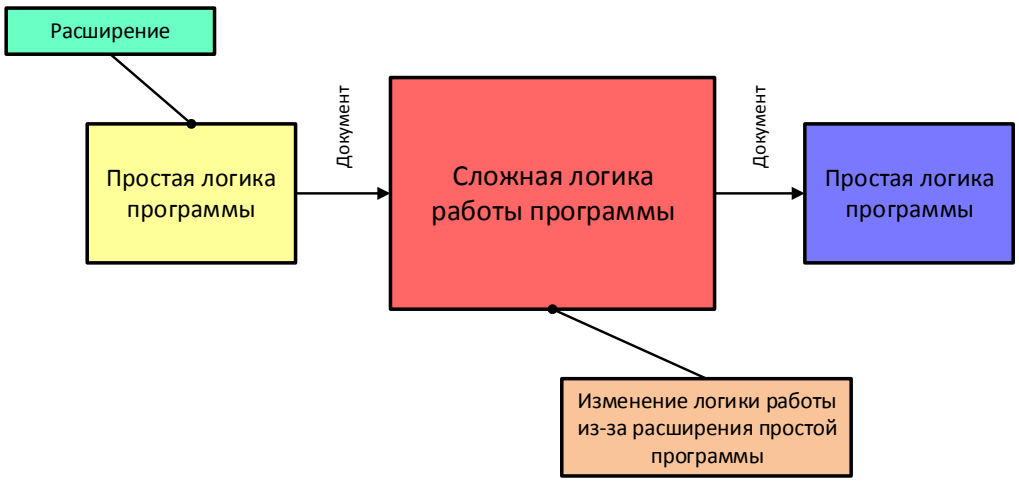


Рис. 31. *Графическая иллюстрация функционального требования по имплементации расширений с максимальным применением стандартных механизмов ERP-систем*

5.6. Типовые ошибки при подготовке функциональных спецификаций

Спецификация на доработку ERP-системы подготовлена и обогащена функциональными требованиями. Далее после ее согласования она отправится в разработку. Однако процесс подтверждения функциональной спецификации может затянуться. Ниже даны ошибки в документе спецификации, усложняющие процесс согласования:

- отсутствие TO-BE модели и концепции решения, без этих пунктов достаточно сложно объяснить пользователю, в чем состоит предлагаемое решение и как в последующем оно будет применяться в рамках непрерывного бизнес-процесса. К сожалению, детали спецификации не смогут прояснить содержание не техническому специалисту;

- чрезмерное количество допущений, здесь важно понимать, что допущение – это вынужденная мера, когда никто не может подтвердить или опровергнуть гипотезу, сильно влияющую на архитектуру программы. Поэтому число допущений должно быть сведено к минимуму;
- указание алгоритмов выборки данных и заполнения полей без описания, применение SQL-запросов позволяет унифицировать описание логики обработки данных, что весьма кстати разработчику. Однако, не следует забывать, что спецификацию будут читать также и бизнес-пользователи, поэтому каждый алгоритм должен сопровождаться человекопонятным описанием.

Обновленная по результатам подтверждения функциональная спецификация передается ABAP-разработчику для реализации. Плановое время разработки определяется тем же Оценщиком, что применялся для задания продолжительности написания документа спецификации.

6. Дальнейшие шаги после подготовки функциональной спецификации

Реализованная функциональная спецификация подлежит испытанию с использованием V-модели разработки через тестирование (рис.13). Данная модель задает проведение функционально-модульного, системного, интеграционного и непрерывного приемочного видов тестирований, по результатам которых возможны изменения логики работы программы. Подобные изменения отражаются в документе спецификации, для чего применяют цветовую индикацию (рис.32).

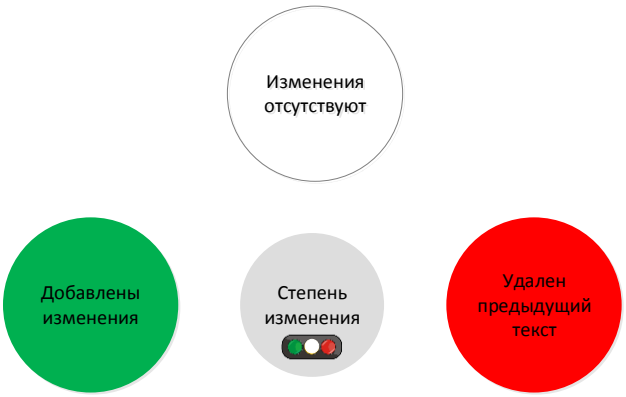


Рис. 32. *Цветовая индикация изменений в функциональной спецификации*

Успешно протестированная и принятая пользователями разработка сопровождается формированием следующего вида спецификации – техническая спецификация.

Определение 11. *Техническая спецификация на разработку (Technical Specification)* – это документ, содержащий программный код разработанного приложения, названия программы или транзакции, структуру разработанных таблиц баз данных, а также подтверждающие копии экранов.

На этом жизненный цикл функциональной спецификации не заканчивается, после завершения периода поддержки продуктивного запуска (Post Go-live support) документ спецификации может обновляться, но уже в рамках процедуры усовершенствования ERP-системы (Continues Improvement) и дальнейшего формирования запросов на изменение (Change Request) [20].

Результаты и их обсуждение

В процессе имплементации корпоративных информационных систем возникает множество сложностей, преодолеть которые помогут заранее известная дорожная карта внедрения, своевременное планирование, отслеживание и корректировка отклонений, а также глубокое погружение в предметную область. Описанная процедура проведения Gap-анализа для последующего формирования документа RTM позволяет единообразно подойти к вопросу оценивания трудозатрат подготовки RICEF-спецификации, реализации разработки на ее основе и проведения модульного испытания приложения.

Выявленные принципы проектирования, уточненные до функциональных требований, позволяют не упустить важные моменты моделирования программ. Предлагаемый подход к формированию концепции решения с использованием обобщенной многоуровневой структуры описания программы позволяет сформировать наглядное верхнеуровневое понимание архитектуры приложения, что значительно упрощает последующее согласование функциональной спецификации с бизнес-пользователями.

Продемонстрированная структура функциональной спецификации носит общий характер и в зависимости от специфики проекта может претерпевать изменения. Введены определения требований, матрицы отслеживания, трудозатрат, видов разработок, допущений и др. позволяют получить расширенное представление о ходе подготовки и реализации спецификации на разработку, являющейся лишь небольшой каплей в море крупных проектов имплементации ERP/ERP2-систем. Дальнейшее развитие работы заключается в детальной проработке способов, методов и методологий внедрения КИС, а также идентификация точек соприкосновения с задачами проектирования и реализации пользовательских программ.

Литература

1. Калянов Г.Н. Консалтинг: от бизнес-стратегии к корпоативной информационно-управляющей системе. – М.: Горячая линия – Телеком, 2014. – 210 с.
2. О'Лири Д. ERP системы. Современное планирование и управление ресурсами предприятия. Выбор, внедрение и эксплуатация / Пер. с англ. Водянова Ю.И. – М.: Вершина, 2004. – 272 с.
3. Кнут Д. Искусство программирования, том 1. Основные алгоритмы. / Пер. с англ. под ред. Козаченко Ю. - М.: Вильямс, 2006. – 720 с.
4. Архангельский А.Я. Программирование в C++ Builder. – М.: Бином, 2010. – 1508 с.
5. Котеров И.В., Симдянов И.В. PHP7. – СПб.: БХВ-Петербург, 2017. – 1088 с.
6. ГОСТ Р 54869-2011. Проектный менеджмент. Требования к управлению проектом. – М.: Стандартинформ, 2011. – 10 с.
7. Zandhuis A., Stellingwerf R. ISO 21500. Guidance on Project Management. A pocket guide. – NL.: Van Haren Publishing, 2013. – 148 p.
8. Степанов Д.Ю. Формирование универсальных требований к пользовательским программам при подготовке спецификации на ABAP-разработку // Актуальные проблемы современной науки. – 2014. – т.78, №4. – с.258-268. – URL: http://stepanovd.com/science/26-article-2014-4-design.
9. Анализ, проектирование и разработка корпоративных информационных систем: уровень приложений [Электронный ресурс] // Официальный сайт Дмитрия Степанова – Режим доступа: https://stepanovd.com/training/12-erp/52-erp-8-applicationlevel (дата обращения 01.09.2019).
10. Степанов Д.Ю. Анализ, проектирование и разработка корпоративных информационных систем: теория и практика // Российский технологический журнал. – 2015. – т.8, №3. – с.227-238. – URL: http://stepanovd.com/science/31-article-2015-2-erpthpr.
11. ANSI/PMI 99-001-2014. A Guide to the Project Management Body of Knowledge (PMBOK Guide). – Pennsylvan.: Project Management Institute, 2013. – 589 p.
12. Степанов Д.Ю. Проблемы внедрения корпоративных информационных систем: уровень приложений // Менеджмент сегодня. – 2015. – т.87, №3. – с.180-191. – URL: http://stepanovd.com/science/30-article-2015-1-erpappl.

13. Гвоздева Т.В., Баллод Б.А. Проектирование информационных систем: учебное пособие. – Ростов н/Д.: Феникс, 2009. – 508 с.
14. Кречмер Р., Вейс В. Разработка приложений SAP R/3 на языке ABAP/4. - М.: Лори. 1998.
15. Ким Д.П. Теория автоматического управления: линейные системы. – М.: ФИЗМАТЛИТ, 2003. – 288 с.
16. Антонов А.В. Системный анализ. – М.: Высшая школа, 2008. – 456 с.
17. Заботина Н.Н. Проектирование информационных систем: учебное пособие. – М.: ИНФРА-М, 2014. – 330 с.
18. Грофф Д.Р., Вайнберг П.Н., Оппель Э.Д. SQL. Полное руководство. – М.: Вильямс, 2014. – 960 с.
19. Brand H. SAP R/3 Implementation With ASAP: The Official SAP Guide. – NJ.: Sybex Inc., 1999. – 591 p.
20. Лодон Дж., Лодон К. Управление информационными системами. / Пер. с англ. под ред. Трутнева Д.Р. - СПб.: Питер, 2005. - 912 с.

Выходные данные статьи

Степанов Д.Ю. Подготовка функциональных спецификаций для разработки корпоративных информационных систем на примере SAP ERP (часть 2). – 2019. – №4(8). – С. 1-18. – URL: https://corpinfosys.ru/archive/issue-8/69-2019-8-functionalspec.

Об авторе

Степанов Дмитрий Юрьевич – кандидат технических наук, доцент МИРЭА, принимал участие более чем в 10 проектах внедрения корпоративных информационных систем на базе SAP, Microsoft и Sage. Специализируется на управлении материальными потоками, сбытом и системой документов. Автор более 25 статей, в том числе в «Логистика сегодня», «Проблемы экономики», «САПер». Электронный адрес: mail@stepanovd.com.