


МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ЧИТИНСКИЙ ИНСТИТУТ
КОЛЛЕДЖ

УТВЕРЖДАЮ:
Первый заместитель директора

Н.В. Раевский
«25» июня 2024 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ПО ПМ 02
ОСУЩЕСТВЛЕНИЕ ИНТЕГРАЦИИ ПРОГРАММНЫХ МОДУЛЕЙ
МДК.02.01. Технология разработки программного обеспечения
09.02.07 Информационные системы и программирование

Чита 2024

Структура фонда оценочных средств
МДК.02.01. Технология разработки программного обеспечения
по специальности 09.02.07 Информационные системы и программирование

Код компетенции	Умения и знания	Основные показатели оценки результата	Методы оценки
Умения:			
ОК01-07, ОК09 ПК 2.1. ПК 2.2 ПК 2.3	Знания: Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию программных модулей. Виды и варианты интеграционных решений. Современные технологии и инструменты интеграции. Основные протоколы доступа к данным. Методы и способы идентификации сбоев и ошибок при интеграции приложений. Методы отладочных классов. Стандарты качества программной документации. Основы организации инспектирования и верификации. Встроенные и основные специализированные инструменты анализа качества программных продуктов. Графические средства проектирования архитектуры программных продуктов. Методы организации работы в команде разработчиков.	Тестирование	76% правильных ответов
	Умения: Анализировать проектную и техническую документацию. Использовать специализированные графические средства построения и анализа архитектуры программных продуктов. Организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов. Определять источники и приемники данных. Проводить сравнительный анализ. Выполнять отладку, используя методы и инструменты условной компиляции (классы Debug и Trace). Оценивать размер минимального набора тестов. Разрабатывать тестовые пакеты и тестовые сценарии.	Практическое задание	Экспертное наблюдение за ходом выполнения практического задания, результат выполнения практической работы не менее 76%

	Выявлять ошибки в системных компонентах на основе спецификаций.		
ПК 2.4.	<p>Знания: Модели процесса разработки программного обеспечения.</p> <p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Основы верификации и аттестации программного обеспечения.</p> <p>Методы и способы идентификации сбоев и ошибок при интеграции приложений.</p> <p>Методы и схемы обработки исключительных ситуаций.</p> <p>Основные методы и виды тестирования программных продуктов.</p> <p>Приемы работы с инструментальными средствами тестирования и отладки.</p> <p>Стандарты качества программной документации.</p> <p>Основы организации инспектирования и верификации.</p> <p>Встроенные и основные специализированные инструменты анализа качества программных продуктов.</p> <p>Методы организации работы в команде разработчиков.</p>	Тестирование	76% правильных ответов
	<p>Умения:</p> <p>Использовать выбранную систему контроля версий.</p> <p>Анализировать проектную и техническую документацию.</p> <p>Выполнять тестирование интеграции.</p> <p>Организовывать постобработку данных.</p> <p>Использовать приемы работы в системах контроля версий.</p> <p>Оценивать размер минимального набора тестов.</p> <p>Разрабатывать тестовые пакеты и тестовые сценарии.</p> <p>Выполнять ручное и автоматизированное тестирование программного модуля.</p> <p>Выявлять ошибки в системных компонентах на основе спецификаций.</p>	Лабораторные работы	Экспертное наблюдение за ходом выполнения лабораторной работы, результат выполнения работы не менее 76%
ПК 2.5.	<p>Знания:</p> <p>Модели процесса разработки программного обеспечения.</p> <p>Основные принципы процесса разработки программного обеспечения.</p> <p>Основные подходы к интегрированию программных модулей.</p> <p>Основы верификации и аттестации программного обеспечения.</p>	Тестирование	76% правильных ответов

	<p>Стандарты качества программной документации.</p> <p>Основы организации инспектирования и верификации.</p> <p>Встроенные и основные специализированные инструменты анализа качества программных продуктов.</p> <p>Методы организации работы в команде разработчиков.</p>		
	<p>Умения:</p> <p>Использовать выбранную систему контроля версий.</p> <p>Использовать методы для получения кода с заданной функциональностью и степенью качества.</p> <p>Анализировать проектную и техническую документацию.</p> <p>Организовывать постобработку данных.</p> <p>Приемы работы в системах контроля версий.</p> <p>Выявлять ошибки в системных компонентах на основе спецификаций.</p>	Лабораторные работы	<p>Экспертное наблюдение за ходом выполнения лабораторной работы, результат выполнения работы не менее 76%</p>

Структура фонда оценочных средств по МДК.02.01. Технология разработки программного обеспечения

№п/п	Проверяемые умения, знания, ОК, ПК	Тема из рабочей программы	Наименование оценочного средства	Критерии оценивания
1.	ПК 2.1, ПК 2.4, ПК 2.5	Основные понятия и стандартизация требований к программному обеспечению	Практическое занятие «Анализ предметной области» Практическое занятие «Разработка и оформление технического задания»	0-4 - «2» 5-6 - «3» 7-8 - «4» 9-10 - «5»
2.	ПК 2.1, ПК 2.4, ПК 2.5	Описание и анализ требований. Диаграммы IDEF	Лабораторная работа «Построение диаграммы Вариантов использования и диаграммы Последовательности» Лабораторная работа «Построение диаграммы Деятельности, диаграммы Состояний и диаграммы Классов» Лабораторная работа «Построение диаграммы компонентов» Лабораторная работа «Построение диаграмм потоков данных»	0-4 - «2» 5-6 - «3» 7-8 - «4» 9-10 - «5»
3	ПК 2.1, ПК 2.4, ПК 2.5	Оценка качества программных средств	Лабораторная работа «Разработка тестового сценария» Лабораторная работа «Разработка тестовых пакетов» Лабораторная работа «Инспекция программного кода на предмет соответствия стандартам кодирования»	0-10 - «2» 11-14 - «3» 15-17 - «4» 18-20 - «5»

4			Устный зачет	0-7 - «2» 8-10 - «3» 11-13 - «4» 14-15 - «5» «5» – верный, подробный ответ; «4» – одна ошибка или не более двух недочетов; «3» – две ошибки или не более трех недочетов; «2» – больше двух ошибок или трех недочетов
				0-10 - «2» 11-14 - «3» 15-17 - «4» 18-20 - «5»
6		Итоговая аттестация за 2 семестр	Устный зачет	«5» – верный, подробный ответ; «4» – одна ошибка или не более двух недочетов; «3» – две ошибки или не более трех недочетов; «2» – больше двух ошибок или трех недочетов.
7		Итоговая аттестация по ПМ	Экзамен	«5» – верный, подробный ответ; «4» – одна ошибка или не более двух недочетов; «3» – две ошибки или не более трех недочетов; «2» – больше двух ошибок или трех недочетов

Практическое занятие «Анализ предметной области»

Цель работы: описать и проанализировать ИС, определить необходимые элементы КТС ИС и системного ПО ИС.

Теоретические сведения

Проблемы управления программными проектами впервые появились в 60-х– начале 70-х годов прошлого века, когда провалились многие большие проекты по разработке программных продуктов. Были зафиксированы задержки в создании ПО, программное обеспечение было ненадежным, затраты на разработку в несколько раз превосходили первоначальные оценки и т.д. Провалы этих проектов обуславливались не только некомпетентностью руководителей и программистов. Напротив, в этих больших поисковых проектах принимали участие люди, уровень квалификации которых был явно выше среднего. Причины провалов коренились в тех подходах, которые использовались в управлении проектами. Применяемая методика была основана на опыте управления техническими проектами и оказалась неэффективной при разработке программных проектов.

Руководители программных проектов выполняют такую же работу, что и руководители технических проектов. Вместе с тем процесс разработки ПО существенно отличается от процессов реализации технических проектов. Ниже приведен небольшой список этих отличий.

1. Программный продукт нематериален. Менеджер судостроительного проекта или проекта постройки здания видит результат выполнения своего проекта. Если реализация проекта отстает от графика, то это видно по незавершенности конструкции. В противоположность этому процент незавершенности программного проекта нельзя увидеть или потрогать. Менеджер программного проекта может полагаться только на документацию, которая фиксирует процесс разработки программного продукта.

2. Не существует стандартных процессов разработки программного обеспечения. На сегодняшний день не существует четкой зависимости между процессом создания ПО и типом создаваемого программного продукта. Другие технические дисциплины имеют длительную историю, процессы разработки технических изделий многократно опробованы и проверены. Изучением же процессов создания ПО специалисты занимаются только последние несколько лет. Поэтому прока нельзя точно предсказать, на каком этапе процесса разработки ПО могут возникнуть проблемы, угрожающие всему проекту.

3. Большие программные проекты – это часто одноразовые проекты. Большие программные проекты, как правило, значительно отличаются от проектов, реализованных ранее. Поэтому, чтобы уменьшить неопределенность в планировании проекта, руководители проектов должны обладать очень большим практическим опытом. Но постоянные технологические изменения в компьютерной технике обесценивают предыдущий опыт. Перечисленные особенности могут привести к тому, что реализация проекта выйдет за рамки временного графика или бюджетных ассигнований. Об этом всегда нужно помнить.

Процессы управления программными проектами

Невозможно описать и стандартизировать все работы, выполняемые менеджером проекта по созданию ПО, но в большинстве случаев к ним относятся.

- Написание предложений по созданию ПО.
- Планирование и составление графика работ проекта.
- Оценивание стоимости проекта.
- Контроль процессов выполнения работ.
- Подбор персонала.
- Написание отчетов и представлений.

Время выполнения больших программных проектов может занимать несколько лет. В течение этого времени цели и намерения организации, оказавшей программный проект, мо-

гут существенно измениться. Может оказаться, что разрабатываемый программный продукт стал уже ненужным либо исходные требования к ПО устарели и их нужно кардинально менять. В такой ситуации руководство организации-разработчика может принять решение о прекращении разработки ПО или об изменении проекта в целом.

Планирование проекта

Эффективное управление проектами напрямую зависит от правильного планирования работ. План, разработанный на начальном этапе проекта, рассматривается всеми его участниками как руководящий документ, выполнение которого должно привести к успешному завершению проекта. Этот первоначальный план должен максимально подробно описывать все этапы реализации проекта.

План проекта должен показать ресурсы, необходимые для реализации проекта, разделение работ на этапы и временной график выполнения этих этапов. Детализация планов проектов очень разнится в зависимости от типа разрабатываемого программного продукта и организации-Разработчика.

Но в любом случае большинство планов содержит следующие разделы.

1. Введение. Краткое описание целей проекта и проектных ограничений (бюджетных, временных и т.д.).

2. Организация выполнения проекта. Описание способа подбора команды разработчиков и распределение обязанностей между членами команды.

3. Анализ рисков. Описание возможных проектных рисков, вероятность их проявления и стратегий, направленных на их уменьшение.

4. Аппаратные и программные ресурсы для реализации проекта. Перечень аппаратных средств и программного обеспечения, необходимого для разработки программного продукта.

5. Разбиение работ на этапы. Проект разбивается на отдельные процессы, определяются этапы выполнения проекта, приводится описание результатов каждого этапа и контрольные отметки.

6. График работ. В графике работ отображаются зависимости между отдельными этапами разработки по, оценки времени их выполнения и распределение членов команды проекта по отдельным этапам.

7. Механизмы контроля и мониторинга за ходом выполнения проекта. Описываются механизмы и сроки предоставления отчетов о ходе работ, а также механизмы мониторинга всего проекта.

При планировании проекта разработки ПО определяются контрольные точки – вехи, отмечающие окончание определенного этапа работ. Для каждой вехи создается отчет, который предоставляется руководству проекта.

При определении контрольных точек весь процесс создания ПО должен быть разбит на отдельные этапы с указанным «выходом» (результатом) каждого этапа. Например, на рис. 1 показаны этапы разработки спецификации требований в случае, когда для ее проверки используется прототип системы.



Рисунок 1. Этапы процесса разработки спецификации

Информационный процесс - это осуществление всей совокупности следующих элементарных информационных актов: прием или создание информации, ее хранение, передача и использование. Информационная система - это совокупность механизмов, обеспечивающих полное осуществление информационного процесса.

Вне ИС информация может лишь сохраняться в виде записей на тех или иных физических носителях, но не может быть ни принятой, ни переданной, ни использованной.

Информационная система - организационно-техническая система, которая предназначена для выполнения информационно-вычислительных работ или предоставления информационно-вычислительных работ или предоставления информационно-вычислительных услуг, удовлетворяющих потребности системы управления и ее пользователей – управленческого персонала, внешних пользователей путем использования и/или создания информационных продуктов. Информационные системы существуют в рамках системы управления и полностью подчинены целям функционирования этих систем.

Информационно-вычислительная работа - деятельность, связанная с использованием информационных продуктов. Типичным примером информационной работы является поддержка информационных технологий управления.

Информационно-вычислительная услуга - это разовая информационно-вычислительная работа. Под информационным продуктом понимается вещественный или нематериальный результат интеллектуального человеческого труда, обычно материализованный на определенном носителе, например разнообразных программных продуктов, выходной информации в виде документов управления, баз данных, хранилищ данных, баз знаний, проектов ИС и ИТ.

Методологическую основу изучения ИС составляет системный подход, в соответствии с которым любая система представляет собой совокупность взаимосвязанных объектов, функционирующих совместно для достижения общей цели.

Информационная система представляет собой совокупность функциональной структуры, информационного, математического, технического, организационного и кадрового обеспечения, которые объединены в единую систему в целях сбора, хранения, обработки и выдачи необходимой информации для выполнения функций управления. Она обеспечивает информационные потоки:

I-1 - информационный поток из внешней среды в систему управления, который, с одной стороны, представляет собой поток нормативной информации, создаваемый государственными учреждениями в части законодательства, а с другой стороны - поток информации о конъюнктуре рынка, создаваемый конкурентами, потребителями, поставщиками;

I-2 - информационный поток из системы управления во внешнюю среду (отчетная информация, прежде всего финансовая в государственные органы, инвесторам, кредиторам, потребителям; маркетинговая информация потенциальным потребителям);

I-3 - информационный поток из системы управления на объект, представляет собой совокупность плановой, нормативной и распорядительной информации для осуществления хозяйственных процессов;

I-4 - информационный поток от объекта в систему управления, который отражает учетную информацию о состоянии объекта управления экономической системой (сырья, материалов, денежных, энергетических, трудовых ресурсов, готовой продукции и выполненных услугах) в результате выполнения хозяйственных процессов.

Задачи информационных систем

Корпоративные системы позволяют решить следующие задачи:

- гарантировать требуемое качество управления предприятием;
- повысить оперативность и эффективность взаимодействия между подразделениями;
- обеспечить управляемость качеством выпускаемой продукции;
- увеличить экономическую эффективность деятельности предприятия;
- создать систему статистического учета на предприятии;
- осуществлять прогноз развития предприятия;
- создать систему стратегического и оперативного планирования, систему прогнозирования.

Задания для практической работы

1. Выберите предметную область
 2. Выберите название ИС в рамках предметной области.
 3. Определите цель ИС
 4. Проведите анализ осуществимости ИС
 - 4.1. Что произойдет с организацией, если система не будет введена в эксплуатацию?
 - 4.2. Какие текущие проблемы существуют в организации и как новая система поможет их решить?
 - 4.3. Каким образом (и будет ли) ИС способствовать целям бизнеса?
 - 4.4. Требуется ли разработка ИС технологии, которая до этого раньше не использовалась в организации?
 5. Где будет размещена ИС? Кто является пользователем ИС?
 6. Комплекс технических средств ИТ
 - 6.1. Какие средства компьютерной техники необходимы для ИС?
 - 6.2. Какие средства коммуникационной техники необходимы для ИС?
 - 6.3. Какие средства организационной техники необходимы для ИС?
 - 6.4. Какие средства оперативной полиграфии необходимы для ИС?
 7. Опишите системное ПО ИТ.
- Таблица 1. Варианты предметных областей

Предметная область	Сущность задачи
Страховая медицинская компания	Страховая медицинская компания (СМК) заключает договоры добровольного медицинского страхования с населением и договоры с лечебными учреждениями на лечение застрахованных клиентов. При возникновении страхового случая клиент подает заявку на оказание медицинских услуг по условиям договора инспектору, который работает с данным клиентом. Инспектор направляет данного клиента в лечебное учреждение. Отчеты о своей деятельности инспектор предоставляет в бухгалтерию. Бухгалтерия проверяет оплату договоров, перечисляет денежные средства за оказанные услуги лечебным учреждениям, производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики. СМК не только оплачивает лечение застрахованного лица при возникновении с ним страхового случая, но и, при возникновении каких-либо осложнений после лечения, оплачивает лечение этих осложнений
Агентство недвижимости	Агентство недвижимости занимается покупкой, продажей, сдачей в аренду объектов недвижимости по договорам с их собственниками. Агентство управляет объектами недвижимости как физических, так и юридических лиц. Собственник может иметь несколько объектов. В случае покупки или аренды клиент может произвести осмотр объекта. В качестве одной из услуг, предлагаемых агентством, является проведение инспектирования текущего состояния объекта для адекватного определения его рыночной цены. По результатам своей деятельности агентство производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики
Кадровое агентство	Кадровое агентство способствует трудоустройству безработных граждан. Агентство ведет учет и классификацию данных о безработных на основании резюме от них. От предприятий города поступают данные о свободных вакансиях, на основании которых агентство предлагает различные варианты трудоустройства соискателям. В случае положительного исхода поиска вакансии считается заполненной, а безработный становится трудоустроенным. По результатам своей деятельности кадровое агентство производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики
Компания по разработке программных продуктов	Компания заключает договор с клиентом на разработку программного продукта согласно техническому заданию. После утверждения технического задания определяется состав и объем работ, составляется предварительная смета. На каждый проект назначается ответственный за его выполнение – куратор проекта, который распределяет нагрузку между программистами и следит за выполнением технического задания. Когда программный продукт готов, то его внедряют, производят обучение клиента и осуществляют дальнейшее сопровождение. По результатам своей деятельности компания производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики

Туроператор	Туроператор предоставляет возможность своим клиентам осуществить туристическую или деловую поездку в различные города России и мира. При разработке нового тура сначала анализируется текущая ситуация на рынке туризма и выбирается направление тура. После этого определяется статус тура, бронируются места в гостиницах и билеты на переезд к месту тура, разрабатывается культурная/ деловая/ развлекательная программа, утверждаются сроки тура. На каждый тур назначается ответственное лицо от туроператора, которое будет вести данный тур для улаживания проблем в случае возникновения каких-нибудь чрезвычайных или форс-мажорных ситуаций. Клиент приходит в офис туроператора, где вместе с менеджером выбирает уже разработанный тур и оформляет путевку. После возвращения из тура клиент может высказать свои замечания или пожелания, которые будут учтены при доработке существующих туров или при разработке новых. Также, для дальнейшего улучшения тура, туроператор проводит анализ отчетов от посредников (гостиница, гиды и т.д.). По результатам своей деятельности туроператор производит отчисления в налоговые органы и предоставляет отчетность в органы государственной статистики
-------------	---

Практическое занятие «Разработка и оформление технического задания»

Цель работы: приобретение навыков разработки технического задания на программный продукт, ознакомиться с правилами написания технического задания

Теоретические сведения

Техническое задание (ТЗ, техзадание) - исходный документ для проектирования сооружения или промышленного комплекса, конструирования технического устройства (прибора, машины, системы управления и т. д.), разработки информационных систем, стандартов либо проведения научно-исследовательских работ (НИР).

ТЗ содержит основные технические требования, предъявляемые к сооружению, изделию или услуге и исходные данные для разработки. В ТЗ указываются назначение объекта, область его применения, стадии разработки конструкторской (проектной, технологической, программной и т.п.) документации, её состав, сроки исполнения и т. д., а также особые требования, обусловленные спецификой самого объекта либо условиями его эксплуатации. Как правило, ТЗ составляют на основе анализа результатов предварительных исследований, расчётов и моделирования. Типовые требования к составу и содержанию технического задания приведены в таблице 1.

Таблица 1. Состав и содержание технического задания (ГОСТ 34.602- 89)

№ пп	Раздел	Содержание
1	Общие сведения	<ul style="list-style-type: none"> - полное наименование системы и ее условное обозначение - шифр темы или шифр (номер) договора; - наименование предприятий разработчика и заказчика системы, их реквизиты - перечень документов, на основании которых создается ИС - плановые сроки начала и окончания работ - сведения об источниках и порядке финансирования работ - порядок оформления и предъявления заказчику результатов работ по созданию системы, ее частей и отдельных средств
2	Назначение и цели создания (развития) системы	<ul style="list-style-type: none"> - вид автоматизируемой деятельности - перечень объектов, на которых предполагается использование системы - наименования и требуемые значения технических, технологических, производственно-экономических и др. показателей объекта, которые должны быть достигнуты при внедрении ИС
3	Характеристика объектов автоматизации	<ul style="list-style-type: none"> - краткие сведения об объекте автоматизации - сведения об условиях эксплуатации и характеристиках окружающей среды
4	Требования к системе	<p>Требования к системе в целом:</p> <ul style="list-style-type: none"> - требования к структуре и функционированию системы (перечень подсистем, уровни иерархии, степень централизации, способы информационного обмена, режимы функционирования, взаимодействие со смежными системами, перспективы развития системы) - требования к персоналу (численность пользователей, квалификация, режим работы, порядок подготовки) - показатели назначения (степень приспособляемости системы к изменениям процессов управления и значений параметров) - требования к надежности, безопасности, эргономике, транспортабельности, эксплуатации, техническому обслуживанию и ремонту, защите и сохранности

		<p>информации, защите от внешних воздействий, к патентной чистоте, по стандартизации и унификации</p> <p>Требования к функциям (по подсистемам) :</p> <ul style="list-style-type: none"> - перечень подлежащих автоматизации задач - временной регламент реализации каждой функции - требования к качеству реализации каждой функции, к форме представления выходной информации, характеристики точности, достоверности выдачи результатов - перечень и критерии отказов <p>Требования к видам обеспечения:</p> <ul style="list-style-type: none"> - математическому (состав и область применения мат. моделей и методов, типовых и разрабатываемых алгоритмов) - информационному (состав, структура и организация данных, обмен данными между компонентами системы, информационная совместимость со смежными системами, используемые классификаторы, СУБД, контроль данных и ведение информационных массивов, процедуры придания юридической силы выходным документам) - лингвистическому (языки программирования, языки взаимодействия пользователей с системой, системы кодирования, языки ввода- вывода) - программному (независимость программных средств от платформы, качество программных средств и способы его контроля, использование фондов алгоритмов и программ) - техническому - метрологическому - организационному (структура и функции эксплуатирующих подразделений, защита от ошибочных действий персонала) - методическому (состав нормативно- технической документации)
5	Состав и содержание работ по созданию системы	<ul style="list-style-type: none"> - перечень стадий и этапов работ - сроки исполнения - состав организаций — исполнителей работ - вид и порядок экспертизы технической документации - программа обеспечения надежности - программа метрологического обеспечения
6	Порядок контроля и приемки системы	<ul style="list-style-type: none"> - виды, состав, объем и методы испытаний системы - общие требования к приемке работ по стадиям - статус приемной комиссии
7	Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие	<ul style="list-style-type: none"> - преобразование входной информации к машиночитаемому виду - изменения в объекте автоматизации - сроки и порядок комплектования и обучения персонала
8	Требования к документированию	<ul style="list-style-type: none"> - перечень подлежащих разработке документов - перечень документов на машинных носителях
9	Источники разработки	<ul style="list-style-type: none"> - документы и информационные материалы, на основании которых разрабатывается ТЗ и система

Порядок разработки технического задания

Разработка технического задания выполняется в следующей последовательности. Прежде всего, устанавливают набор выполняемых функций, а также перечень и характеристики исходных данных.

Затем определяют перечень результатов, их характеристики и способы представления.

Далее уточняют среду функционирования программного обеспечения: конкретную комплектацию и параметры технических средств, версию используемой операционной системы и, возможно, версии и параметры другого установленного программного обеспечения, с которым предстоит взаимодействовать будущему программному продукту.

В случаях, когда разрабатываемое программное обеспечение собирает и хранит некоторую информацию или включается в управление каким-либо техническим процессом, необходимо также четко регламентировать действия программы в случае сбоев оборудования и энергоснабжения.

1. Общие положения

1.1 Техническое задание оформляют в соответствии с ГОСТ 19.106-78 на листах формата А4 и А3 по ГОСТ 2.301-68, как правило, без заполнения полей листа. Номера листов (страниц) проставляют в верхней части листа над текстом.

1.2 Лист утверждения и титульный лист оформляют в соответствии с ГОСТ 19.104-78. Информационную часть (аннотацию и содержание), лист регистрации изменений допускаются и в документ не включать.

1.3 Для внесения изменений и дополнений в техническое задание на последующих стадиях разработки программы или программного изделия выпускают дополнение к нему. Согласование и утверждение дополнения к техническому заданию проводят в том же порядке, который установлен для технического задания.

1.4. Техническое задание должно содержать следующие разделы:

- введение;
- наименование и область применения;
- основание для разработки;
- назначение разработки;
- технические требования к программе или программному изделию;
- технико-экономические показатели;
- стадии и этапы разработки;
- порядок контроля и приемки;
- приложения.

В зависимости от особенностей программы или программного изделия допускается уточнять содержание разделов, вводить новые разделы или объединять отдельные из них. При необходимости допускается в техническое задание включать приложения.

2. Содержание разделов

2.1 Введение должно включать краткую характеристику области применения программы или программного продукта, а также объекта (например, системы), в котором предполагается их использовать. Основное назначение введения - продемонстрировать актуальность данной разработки и показать, какое место эта разработка занимает в ряду подобных.

2.2 В разделе «Наименование и область применения» указывают наименование, краткую характеристику области применения программы или программного изделия и объекта, в котором используют программу или программное изделие.

2.3 В разделе «Основание для разработки» должны быть указаны:

- документ (документы), на основании которых ведется разработка. Таким документом может служить план, приказ, договор и т. п.;
- организация, утвердившая этот документ, и дата его утверждения;
- наименование и (или) условное обозначение темы разработки.

2.4 В разделе «Назначение разработки» должно быть указано функциональное и эксплуатационное назначение программы или программного изделия.

2.5 Раздел «Технические требования к программе или программному изделию» должен содержать следующие подразделы:

- требования к функциональным характеристикам;
- требования к надежности;
- условия эксплуатации;
- требования к составу и параметрам технических средств;
- требования к информационной и программной совместимости;
- требования к маркировке и упаковке;
- требования к транспортированию и хранению;
- специальные требования.

В подразделе «Требования к функциональным характеристикам» должны быть указаны требования к составу выполняемых функций, организации входных и выходных данных, временным характеристикам и т. п.

2.5.2 В подразделе «Требования к надежности» должны быть указаны требования к обеспечению надежного функционирования (обеспечение устойчивого функционирования, контроль входной и выходной информации, время восстановления после отказа и т. п.).

2.5.3 В подразделе «Условия эксплуатации» должны быть указаны условия эксплуатации (температура окружающего воздуха, относительная влажность и т. п. для выбранных типов носителей данных), при которых должны обеспечиваться заданные характеристики, а также вид обслуживания, необходимое количество и квалификация персонала.

2.5.4 В подразделе «Требования к составу и параметрам технических средств» указывают необходимый состав технических средств с указанием их технических характеристик.

2.5.5 В подразделе «Требования к информационной и программной совместимости» должны быть указаны требования к информационным структурам на входе и выходе и методам решения, исходным кодам, языкам программирования. При необходимости должна обеспечиваться защита информации и программ.

2.5.6 В подразделе «Требования к маркировке и упаковке» в общем случае указывают требования к маркировке программного изделия, варианты и способы упаковки.

2.5.7 В подразделе «Требования к транспортированию и хранению» должны быть указаны для программного изделия условия транспортирования, места хранения, условия хранения, условия складирования, сроки хранения в различных условиях.

2.5.8 В разделе «Технико-экономические показатели» должны быть указаны: ориентировочная экономическая эффективность, предполагаемая годовая потребность, экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами.

2.6 В разделе «Стадии и этапы разработки» устанавливают необходимые стадии разработки, этапы и содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены), а также, как правило, сроки разработки и определяют исполнителей.

2.7 В разделе «Порядок контроля и приемки» должны быть указаны виды испытаний и общие требования к приемке работы.

2.8 В приложениях к техническому заданию при необходимости приводят:

- перечень научно-исследовательских и других работ, обосновывающих разработку;
- схемы алгоритмов, таблицы, описания, обоснования, расчеты и другие документы, которые могут быть использованы при разработке;
- другие источники разработки.

В случаях, если какие-либо требования, предусмотренные техническим заданием, заказчик не предъявляет, следует в соответствующем месте указать «Требования не предъявляются».

Задания для практической работы

- №1.
1. Разработать техническое задание по варианту выбранному в практической работе
 2. Оформить отчет

Порядок выполнения отчета по практической работе

- MS
1. Разработать техническое задание на программный продукт
 2. Оформить работу в соответствии с ГОСТ 19.106-78. При оформлении использовать Office.
 3. Сдать и защитить работу

Защита отчета по практической работе заключается в предъявлении преподавателю полученных результатов (на экране монитора и печатном виде), демонстрации полученных навыков и ответах на вопросы преподавателя.

Лабораторная работа «Построение диаграммы Вариантов использования и диаграммы Последовательности»

Цель работы: исследование процесса построения диаграмм прецедентов и диаграмм взаимодействий в заданной предметной области с помощью пакета Rational Rose

Теоретические сведения

Постановка задачи (описание предметной области).

Магазин осуществляет продажу товаров клиенту путем оформления документов «Заказ». Директор магазина- Антон, принял решение автоматизировать документооборот продаж товара и пригласил для выполнения работ программиста Павла. Поговорив с Антоном, в соответствии с концепцией жизненного цикла (ЖЦ) программы Павел приступил к описанию бизнес процессов, сопровождающих продажу товара. Взяв за основу язык UML, он начал с построения контекстной диаграммы процессов- Use Case diagram. Диаграмма должна ответить на вопрос-«что должно делаться в системе и кто участник этих процессов».

Задание 1. Создание диаграммы вариантов использования и действующих лиц.
Окончательный вид диаграммы показан на рис. 1.

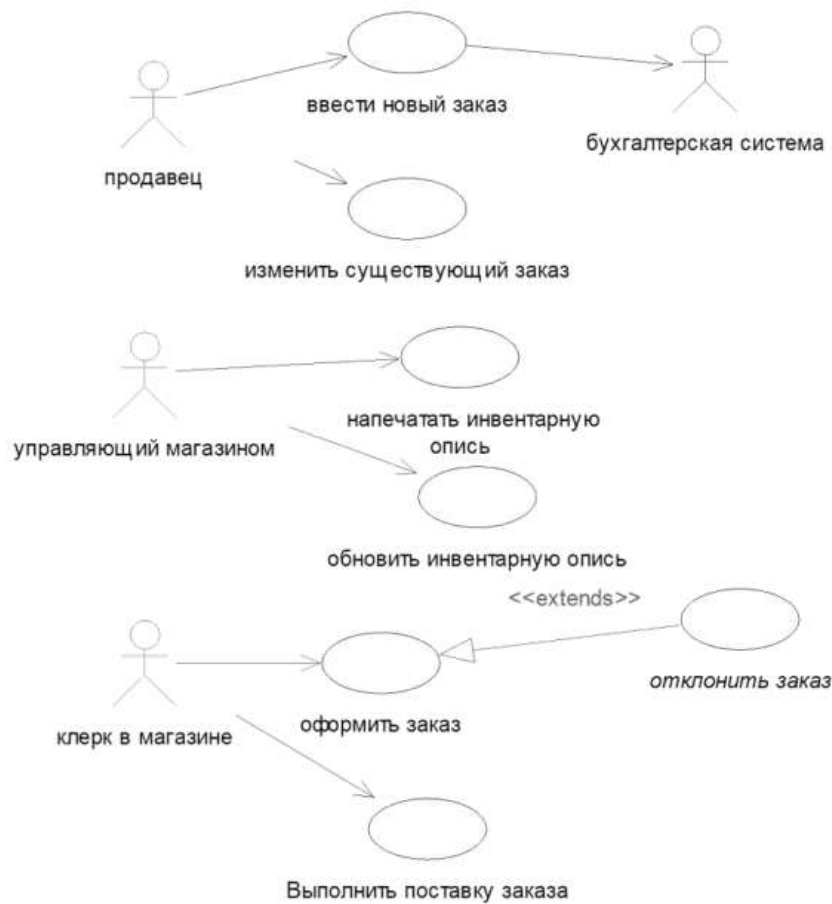


Рис. 1 Диаграмма вариантов использования задачи о заказе товара.

Этапы выполнения

1. Дважды щелкнув мышью на Главной диаграмме Вариантов Использования (Main) в браузере, откройте ее.
2. С помощью кнопки Use Case (Вариант использования) панели инструментов поместите на диаграмму новый вариант использования. Назовите его "Ввести новый заказ".
3. Повторив этапы 2 и 3, поместите на диаграмму остальные варианты использования:
Изменить существующий заказ
Напечатать инвентарную опись
Обновить инвентарную опись
Оформить заказ
Отклонить заказ
Выполнить поставку заказа
4. С помощью кнопки Actor (Действующее лицо) панели инструментов поместите на диаграмму новое действующее лицо.
5. Назовите его "Продавец".
6. Повторив шаги 4 и 5, поместите на диаграмму остальных действующих лиц:
Управляющий магазином
Клерк магазина
Бухгалтерская система
7. Создание абстрактного варианта использования (не требующего дальнейшей декомпозиции).

Щелкните правой кнопкой мыши на варианте использования "Отклонить заказ" на диаграмме.

В открывшемся меню выберите пункт Open Specification (Открыть спецификацию).

Установите флажок Abstract (Абстрактный), чтобы сделать этот вариант использования абстрактным.

Добавление ассоциаций

1. С помощью кнопки Unidirectional Association (Однонаправленная ассоциация) панели инструмента нарисуйте ассоциацию между действующим лицом Продавец и вариантом использования "Ввести заказ".

2. Повторив шаг 1, поместите на диаграмму остальные ассоциации, согласно рис. 1.

Добавление связи расширения

С помощью кнопки Generalization (Обобщение) панели инструментов нарисуйте связь между вариантом использования "Отклонить заказ" и вариантом использования "Оформить заказ". Стрелка должна быть направлена от первого варианта использования ко второму. Связь расширения означает, что вариант использования "Отклонить заказ" при необходимости дополняет функциональные возможности варианта использования "Оформить заказ".

Щелкните правой кнопкой мыши на новой связи между вариантами использования "Отклонить заказ" и "Оформить заказ".

В открывшемся меню выберите пункт Open Specification (Открыть спецификацию).

В раскрывающемся списке стереотипов введите слово extends (расширение), затем нажмите ОК.

Надпись «extends» появится на линии данной связи.

Добавление описаний к вариантам использования

Выделите в браузере вариант использования "Ввести новый заказ".

В окне документации введите следующее описание: "Этот вариант использования дает клиенту возможность ввести новый заказ в систему".

С помощью окна документации добавьте описания ко всем остальным вариантам использования.

Добавление описаний к действующему лицу

Выделите в браузере действующее лицо Продавец.

В окне документации введите следующее описание: "Продавец — это служащий, старающийся продать товар".

С помощью окна документации добавьте описания к остальным действующим лицам.

Задание 2. Создание диаграммы Последовательности.

Согласовав основные бизнес процессы с Антоном, Павел приступил к построению модели бизнес - процессов, что бы ответить на вопрос - «как это должно делаться в системе». Для начала он выбрал наиболее важный Вариант использования - «Ввод нового заказа» и построил для него диаграммы взаимодействия.

Диаграммы взаимодействия включают в себя два типа диаграмм - Последовательности и Кооперативную.

Этапы выполнения

Настройка программной среды

1. В меню модели выберите пункт Tools >- Options (Инструменты >- Параметры).

2. Перейдите на вкладку Diagram (Диаграмма).

3. Установите флажки Sequence numbering, Collaboration numbering и Focus of control.

4. Нажмите ОК, чтобы выйти из окна параметров.

Создание диаграммы Последовательности

1. Щелкните правой кнопкой мыши на Логическом представлении браузера.

2. В открывшемся меню выберите пункт New > Sequence Diagram (Создать > Диаграмма Последовательности).

3. Назовите новую диаграмму: Ввод заказа.

4. Дважды щелкнув на этой диаграмме, откройте ее.

Добавление на диаграмму действующего лица и объектов

1. Перетащите действующее лицо Продавец из браузера на диаграмму.

2. Нажмите кнопку Object (Объект) панели инструментов.

3. Щелкните мышью в верхней части диаграммы, чтобы поместить туда новый объект.

4. Назовите объект Выбор варианта заказа.

5. Повторив шаги 3 и 4, поместите на диаграмму объекты:

- Форма деталей заказа

- Заказ No1234

Добавление сообщений на диаграмму

1. На панели инструментов нажмите кнопку Object Message (Сообщение объекта).

2. Проведите мышью от линии жизни действующего лица Продавец к линии жизни объекта Выбор варианта заказа.

3. Выделив сообщение, введите его имя — Создать новый заказ.

4. Повторив шаги 2 и 3, поместите на диаграмму сообщения:

- Открыть форму — между Выбор Варианта Заказа и Форма деталей Заказа

- Ввести номер заказа, заказчика и число заказываемых предметов — между Продавец и Форма Деталей Заказа

- Сохранить заказ — между Продавец и Форма Деталей Заказа

- Создать пустой заказ — между Форма Деталей Заказа и Заказ N1234

- Ввести номер заказа, заказчика и число заказываемых предметов — между Форма Деталей Заказа и Заказ N1234

- Сохранить заказ — между Форма Деталей Заказа и Заказ N1234

Завершен первый этап работы. Готовая диаграмма Последовательности представлена на рис. 2.

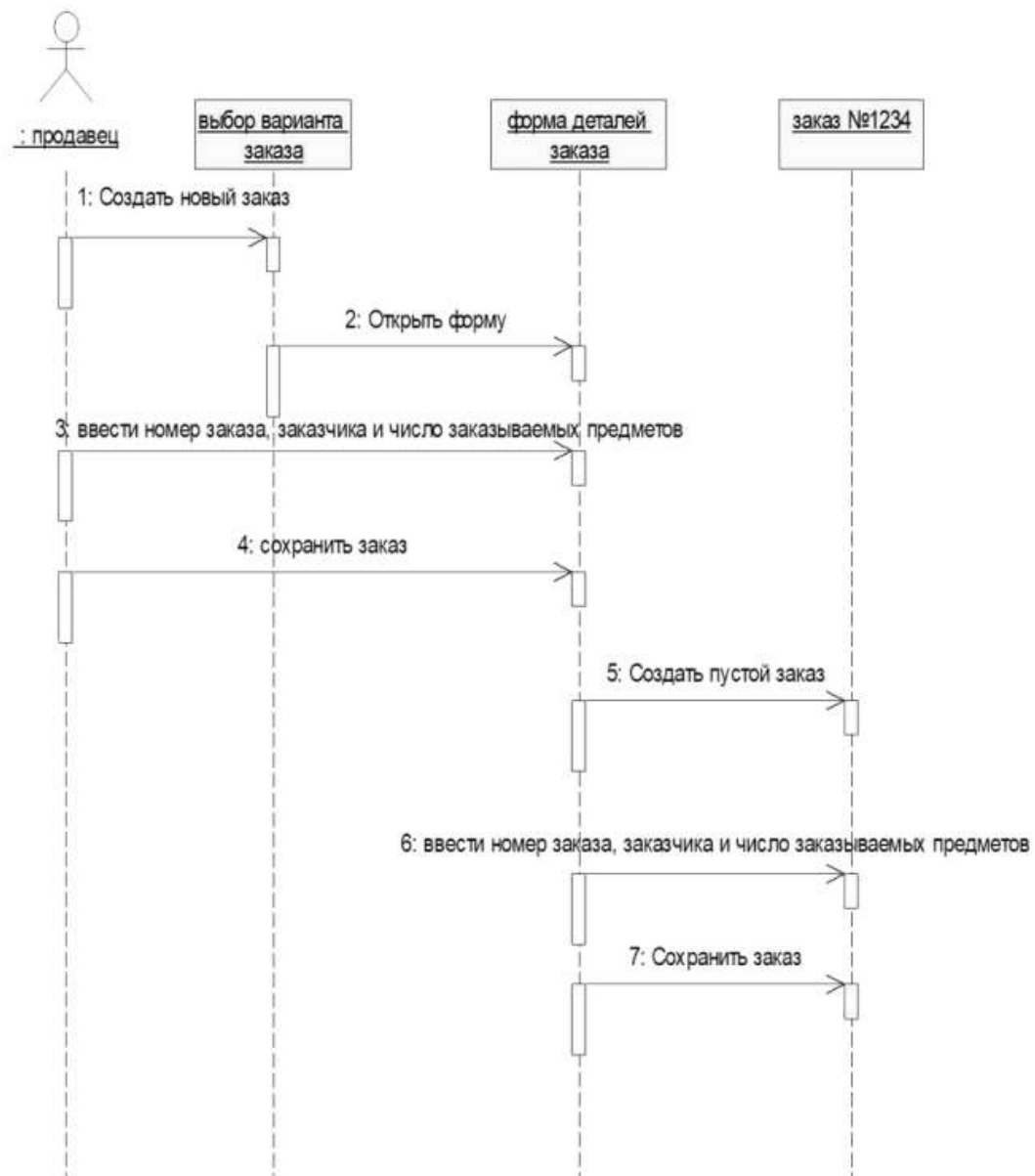


Рис. 2. Диаграмма последовательности без управляющих элементов.

Теперь нужно позаботиться об управляющих объектах и о взаимодействии с базой данных. Как видно из диаграммы, объект Форма Деталей Заказа имеет множество ответственных, с которыми лучше всего мог бы справиться управляющий объект. Кроме того, новый заказ должен сохранять себя в базе данных сам. Вероятно, эту обязанность лучше было бы переложить на другой объект.

Добавление на диаграмму дополнительных объектов

1. Нажмите кнопку Object панели инструментов.
2. Щелкните мышью между объектами Форма Деталей Заказа и Заказ No1234, чтобы поместить туда новый объект.

3. Введите имя объекта — Управляющий заказами.

4. Нажмите кнопку Object панели инструментов.

5. Новый объект расположите справа от Заказ No1234.

6. Введите его имя- Управляющий транзакциями.

Назначение ответственностей объектам

1. Выделите сообщение 5: Создать пустой заказ.
 2. Нажав комбинацию клавиш CTRL+D, удалите это сообщение.
 3. Повторите шаги 1 и 2 для удаления двух последних сообщений:
 - Вести номер заказа, заказчика и число заказываемых предметов
 - Сохранить заказ
 4. Нажмите кнопку Object Message панели инструментов.
 5. Поместите на диаграмму новое сообщение, расположив его под сообщением 4 между Форма деталей заказа и Управляющий заказами.
 6. Назовите его Сохранить заказ.
 7. Повторите шаги 4 — 6, добавив сообщения с шестого по девятое и назвав их:
 - Создать новый заказ — между Управляющий заказами и Заказ No1234
 - Ввести номер заказа, заказчика и число заказываемых предметов- между Управляющий заказами и Заказ No1234
 - Сохранить заказ- между Управляющий заказами и Управляющий транзакциями
 - Информация о заказе — между Управляющий транзакциями и Заказ No1234
 8. На панели инструментов нажмите кнопку Message to Self (Сообщение себе).
 9. Щелкните на линии жизни объекта Управляющий транзакциями ниже сообщения 9, добавив туда рефлексивное сообщение.
 10. Назовите его Сохранить информацию о заказе в базе данных.
- Соотнесение объектов с классами
1. Щелкните правой кнопкой мыши на объекте Выбор варианта заказа.
 2. В открывшемся меню выберите пункт Open Specification (Открыть спецификацию). В раскрывающемся списке классов выберите пункт <New> (Создать). Появится окно спецификации классов.
 4. В поле Name введите Выбор заказа.
 5. Щелкните на кнопке ОК. Вы вернетесь в окно спецификации объекта.
 6. В списке классов выберите класс Выбор Заказа.
 7. Щелкните на кнопке ОК, чтобы вернуться к диаграмме. Теперь объект называется Выбор варианта заказа: Выбор Заказа
 8. Для соотнесения остальных объектов с классами повторите шаги с 1 по 7:
 - Класс Детали заказа соотнесите с объектом Форма деталей заказа
 - Класс Упр_ заказами — с объектом Управляющий заказами
 - Класс Заказ — с объектом Заказ N 1234
 - Класс Упр_ транзакциями — с объектом Управляющий транзакциями
- Соотнесение сообщений с операциями
1. Щелкните правой кнопкой мыши на сообщении 1: Создать новый заказ.
 2. В открывшемся меню выберите пункт <new operation> (создать операцию). Появится окно спецификации операции.
 3. В поле Name введите имя операции —Создать.
 4. Нажмите на кнопку ОК, чтобы закрыть окно спецификации операции и вернуться к диаграмме.
 5. Еще раз щелкните правой кнопкой мыши на сообщении 1.
 6. В открывшемся меню выберите новую операцию Создать().
 7. Повторите шаги с 1 по 6, чтобы соотнести с операциями все остальные сообщения:
 - Сообщение 2: Открыть форму соотнесите с операцией Открыть()
 - Сообщение 3: Ввести номер заказа, заказчика и число заказываемых предметов — с операцией Ввести номер заказа, заказчика и число заказываемых предметов()
 - Сообщение 4: Сохранить заказ — с операцией Сохранить заказ()
 - Сообщение 5: Сохранить заказ — с операцией Сохранить заказ()

- Сообщение 6: Создать пустой заказ – с операцией Создать пустой заказ()
 - Сообщение 7: Ввести номер заказа, заказчика и число заказываемых предметов- с одноименной операцией.
 - Сообщение 8 Сохранить заказ – с операцией Сохранить заказ()
 - Сообщение 9 Информация о заказе – с одноименной операцией
 - Сообщение 10 Сохранить информацию о заказе с одноименной операцией.
- Диаграмма должна выглядеть, как на рис. 3.

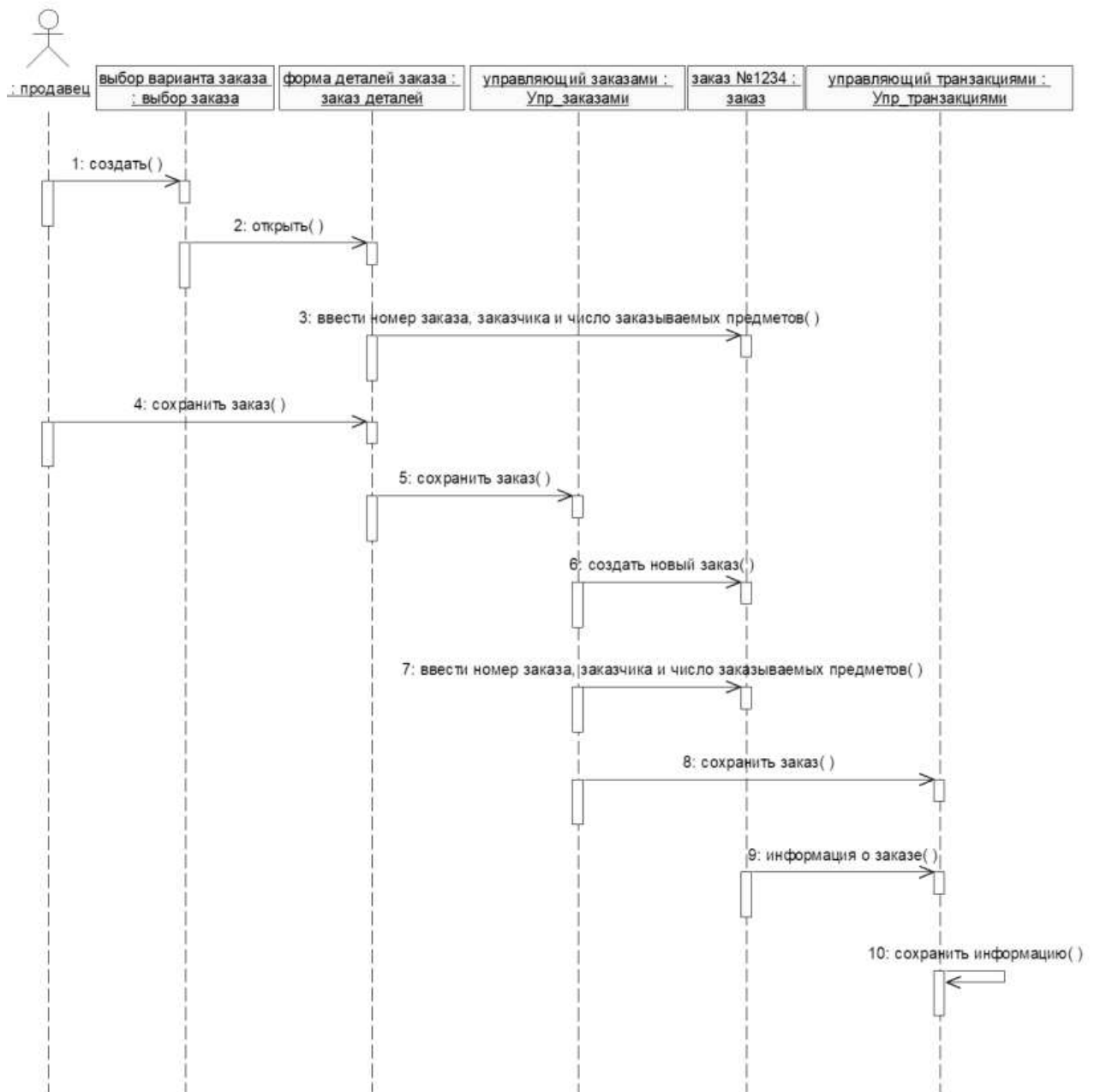


Рис. 3. Окончательный вид диаграммы последовательности

Цель работы: исследование процесса построения диаграммы состояний и диаграммы классов в заданной предметной области с помощью пакета Rational Rose 2000.

Задание 1. Постройте диаграмму Состояний

Постройте диаграмму Состояний для класса Заказ, показанную на рис. 5.



Рис 5. Диаграмма состояний для класса Заказ

Этапы выполнения

Создание диаграммы

1. Найдите в браузере класс Заказ.
2. Щелкните на классе правой кнопкой мыши и в открывшемся меню укажите пункт New >Statechart Diagram (Создать диаграмму состояний).

Добавление начального и конечного состояний

1. Нажмите кнопку Start State (Начальное состояние) панели инструментов.
2. Поместите это состояние на диаграмму.
3. Нажмите кнопку End State (Конечное состояние) панели инструментов.
4. Поместите это состояние на диаграмму.

Добавление суперсостояния

1. Нажмите кнопку State (Состояние) панели инструментов.
2. Поместите это состояние на диаграмму.

Добавление оставшихся состояний

1. На панели инструментов нажмите кнопку State (Состояние).
2. Поместите состояние на диаграмму.
3. Назовите состояние Отменен.
4. На панели инструментов нажмите кнопку State(Состояние).
5. Поместите состояние на диаграмму.
6. Назовите состояние Выполнен.
7. На панели инструментов нажмите кнопку State(Состояние).
8. Поместите состояние на диаграмму внутрь суперсостояния.
9. Назовите состояние Инициализация.
10. На панели инструментов нажмите кнопку State (Состояние).
11. Поместите состояние на диаграмму внутрь суперсостояния.
12. Назовите состояние Выполнение заказа приостановлено.

Описание состояний

1. Дважды щелкните мышью на состоянии Инициализация.
 2. Перейдите на вкладку Detail (Подробно).
 3. Щелкните правой кнопкой мыши в окне Actions(Действия).
 4. В открывшемся меню выберите пункт Insert(Вставить).
 5. Дважды щелкните мышью на новом действии.
 6. Назовите его Сохранить дату заказа.
 7. Убедитесь, что в окне When (Когда) указан пункт On Entry (На входе).
 8. Повторив шаги 3—7, добавьте следующие действия:
 - Собрать клиентскую информацию, в окне When укажите DO (Выполнять между входом и выходом)
 - Добавить к заказу новые позиции, укажите DO
 9. Нажмите два раза на ОК, чтобы закрыть спецификацию.
 10. Дважды щелкните мышью на состоянии Отменен.
 11. Повторив шаги 2—7, добавьте действия:
 - Сохранить дату отмены, укажите On Exit (На выходе)
 12. Нажмите два раза на ОК, чтобы закрыть спецификацию.
 13. Дважды щелкните мышью на состоянии Выполнен.
 14. Повторив шаги 2—7, добавьте действие:
 - Выписать счет, укажите On Exit
 15. Нажмите два раза на ОК, чтобы закрыть спецификацию.
- #### Добавление переходов
1. Нажмите кнопку Transition (Переход) панели инструментов.
 2. Щелкните мышью на начальном состоянии.
 3. Проведите линию перехода к состоянию Инициализация.
 4. Повторив шаги с первого по третий, создайте следующие переходы:
 - От состояния Инициализация к состоянию Выполнение заказа приостановлено
 - От состояния Выполнение заказа приостановлено к состоянию Выполнен
 - От суперсостояния к состоянию Отменен

- От состояния Отменен к конечному состоянию
- От состояния Выполнен к конечному состоянию

5. На панели инструментов нажмите кнопку Transition to Self (Переход к себе).

6. Щелкните мышью на состоянии Выполнение заказа приостановлено

Описание переходов

1. Дважды щелкнув мышью на переходе от состояния Инициализация к состоянию Выполнение заказа приостановлено, откройте окно спецификации перехода.
2. В поле Event (Событие) введите фразу Выполнить заказ.
3. Щелкнув на кнопке ОК, закройте окно спецификации.
4. Повторив шаги с первого по третий, добавьте событие Отменить заказ к переходу между суперсостоянием и состоянием Отменен.
5. Дважды щелкнув мышью на переходе от состояния Выполнение заказа приостановлено к состоянию Выполнен, откройте окно его спецификации.
6. В поле Event (Событие) введите фразу Добавить к заказу новую позицию.
7. Перейдите на вкладку Detail (Подробно).
8. В поле Guard Condition (Сторожевое Условие) введите Не осталось незаполненных позиций.
9. Щелкнув на кнопке ОК, закройте окно спецификации.
10. Дважды щелкните мышью на рефлексивном переходе (Transition to Self) состояния Выполнение заказа приостановлено.
11. В поле Event (Событие) введите фразу Добавить к заказу новую позицию.
12. Перейдите на вкладку Detail (Подробно).
13. В поле Guard Condition (Сторожевое Условие) введите Остаются незаполненные позиции.
14. Щелкнув на кнопке ОК, закройте окно спецификации.

Задание 2. Построение диаграммы Активности для варианта использования «Выполнить поставку Заказа».

Побеседовав с Павлом, Антон понял, что необходимо согласовать логику реализации еще одного варианта использования «Выполнить поставку заказа». Стало ясно, что здесь возможны несколько альтернативных потоков управления. Для таких ситуаций более удобно использовать не диаграммы взаимодействия, приспособленные для единственного потока, а диаграмму активности.

Описание варианта использования.

При оформлении заказа проверяют каждую содержащуюся в нем позицию, чтобы убедиться в наличии соответствующих товаров на складе. После этого выписываются товары для реализации заказа. Во время выполнения этих процедур одновременно проверяется прохождение платежа. Если платеж прошел, и товары имеются на складе, то осуществляется их поставка. Если платеж прошел, но товары на складе отсутствуют, то заказ ставится в ожидание. Если платеж не прошел, то заказ аннулируется.

Этапы выполнения

1. Найдите в браузере вариант использования «Выполнить поставку заказа»

2. Щелкните на классе правой кнопкой мыши и в открывшемся меню укажите пункт New > Activity Diagram (Создать диаграмму активности).

3. Назовите диаграмму «Выполнить поставку» и откройте ее двукратным щелчком мыши

4. На панели инструментов щелкните мышкой на элементе Swimlane, затем на поле диаграммы. На диаграмме появится разделительная линия («водная дорожка»).

5. Установите курсор на заголовок NewSwimlane и нажмите правую клавишу мыши. В выпадающем списке нажмите Select in browser. В браузере выделится этот объект. Нажав правую клавишу мыши в выпадающем списке выберете Open Specification и откройте спецификацию.

Измените поле Name на Клерк. Выберите в поле Class Клерк в магазине.

6. Выполните заново пункты 5-6 и присвойте полю Name Система, Class- Бухгалтерская система.

7. Найдите в браузере сплошной черный кружок (начальное состояние). Перенесите его на дорожку Клерк.

8. Выберете из панели инструментов объект Activity и поместите его на диаграмму в —дорожку Клерк. Измените имя объекта на «Получить заказ».

9. Повторите предыдущий этап, создайте на «дорожке» Клерк 4 новых Activity и присвойте им имена Проверить позицию заказа, закрепить позицию за заказом, Поставить заказ в ожидание, Скомплектовать заказ

10. Поместите на «дорожку» 2 новых объекта End State (конечное состояние). Одному из них измените поле Name на «Выполнить поставку»

11. На дорожку Система поместите новый объект Activity и присвойте полю Name —Проверить платеж. На эту же дорожку поместите новый объект End State и измените в его спецификации поле Name на «Отменить заказ».

12. Поместить на «дорожку» Клерк 2 объекта Horizontal Synchronization (горизонтальная синхронизация). Присвойте полю Name спецификации одного объекта «1», другого- «2».

13. Поместить на «дорожку» Клерк объект Decision (выбор). Через спецификацию присвойте полю Name «Позиция имеется?».

14. Поместить на «дорожку» Система объект Decision. Присвойте полю Name «Деньги поступили?».

15. Щелкните мышкой на панели инструментов объекте- стрелке State Transition (состояние перехода). Затем щелкните мышкой на диаграмме объекта начальное состояние. Удерживая кнопку мыши перенесите курсор на активность «Получить заказ» и лишь затем отпустить курсор. В результате два объекта будут соединены стрелкой.

16. Выполните этап 14, соединив стрелкой объект Активность «Получить заказ» с объектом Horizontal Synchronization 1.

17. Соедините этими же стрелками объекты 1 и «Проверить платеж», 1 и «Проверить позицию заказа», «Проверить заказ» и «Деньги подступили?», «Деньги поступили?» и «Отменить заказ», «Проверить позицию заказа» и «Позиция имеется», «Позиция имеется» и «Закрепить позицию за заказом», «Деньги получены?» и 2, «Закрепить позицию за заказом» и 2, «Позиция имеется?» и «Поставить заказ в ожидание», 2 и «Скомплектовать заказ», «Ском-

плектовать заказ» и «Выполнить поставку», «Поставить заказ в ожидание» и объект Конечное состояние (без имени).

18. Присвоим некоторым стрелкам наименование полю Event (условие перехода). Для этого, установим курсор на стрелке, соединяющей «Деньги получены?» и «Отменить заказ». Двукратным щелчком мыши откроем окно спецификации. В поле Event введем «Нет».

19. Выполним пункт 18 для стрелки, соединяющей «Деньги получены?» и 2 и присвойте Event «Да». Аналогично для стрелки соединяющей «Позиция имеется?» и «Закрепить позицию за заказом» присвоить Event «Да». Стрелке, соединяющей «Позиция имеется?» и «Поставить заказ в ожидание» - «Нет».

20. Добавим элементарные действия (Actions) к активности — Проверить позицию заказа». Установим курсор на «Проверить позицию заказа» и двукратным щелчком мыши откроем окно спецификации. Откроем закладку Actions. Установим курсор на свободное поле и нажмем правую клавишу мыши. В выпадающем меню нажмем Insert. В появившейся заставке в поле When выберем Entry (на входе в активность), В поле Name введем «Просмотреть спецификацию к заказу». Нажать Ok. Вновь нажмем курсор правой мыши и введем новое действие. Полю When присвоим Do (промежуток между входом и выходом), а полю Name «Найти новую позицию». При вводе третьей активности полю When присвойте Exit (выход), а полю Name «Передать результаты поиска».

21. Путем перемещения объектов (установить курсор мыши- нажать- тащить- отпустить) привести диаграмму к виду, показанному на рис. 6.

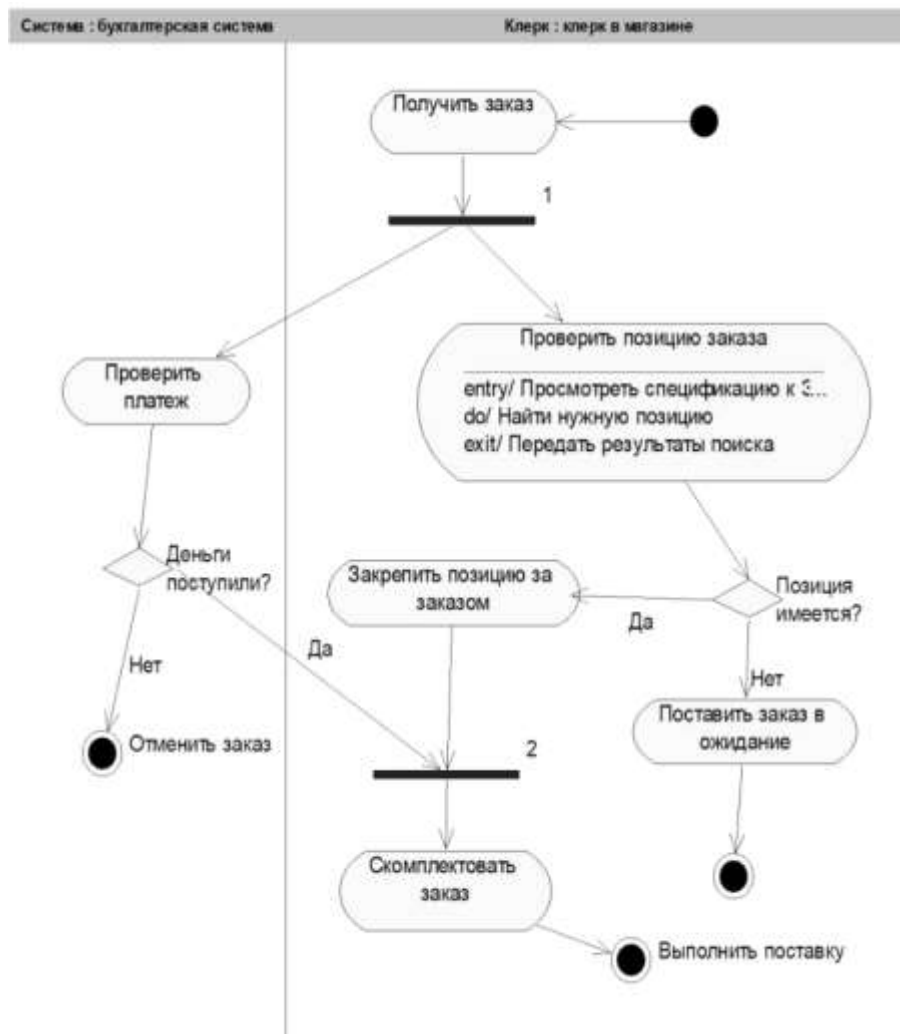


Рис. 6 Диаграмма активности для варианта использования «Выполнить поставку заказа»

Задание 3. Пакеты и классы

В этом упражнении необходимо сгруппировать в пакеты классы, созданные при выполнении предыдущих работ. Затем нужно будет построить несколько диаграмм Классов и показать на них классы и пакеты системы.

Создание диаграммы Классов

Объедините обнаруженные классы в пакеты. Создайте диаграмму Классов для отображения пакетов, диаграммы Классов, для представления классов в каждом пакете и диаграмму Классов для представления всех классов варианта использования "Ввести новый заказ".

Этапы выполнения

Создание пакетов

1. Щелкните правой кнопкой мыши на Логическом представлении браузера.
2. В открывшемся меню выберите пункт New > Package (Создать > Пакет).
3. Назовите новый пакет Сущности.
4. Повторив шаги 1—3, создайте пакеты Границы и Управление.

Создание Главной диаграммы Классов

1. Дважды щелкнув мышью на Главной диаграмме Классов, находящейся под Логическим представлением браузера, откройте ее.
 2. Перетащите пакет Сущности из браузера на диаграмму.
 3. Перетащите пакеты Границы и Управление из браузера на диаграмму.
- Главная диаграмма Классов должна выглядеть, как показано на рис. 7

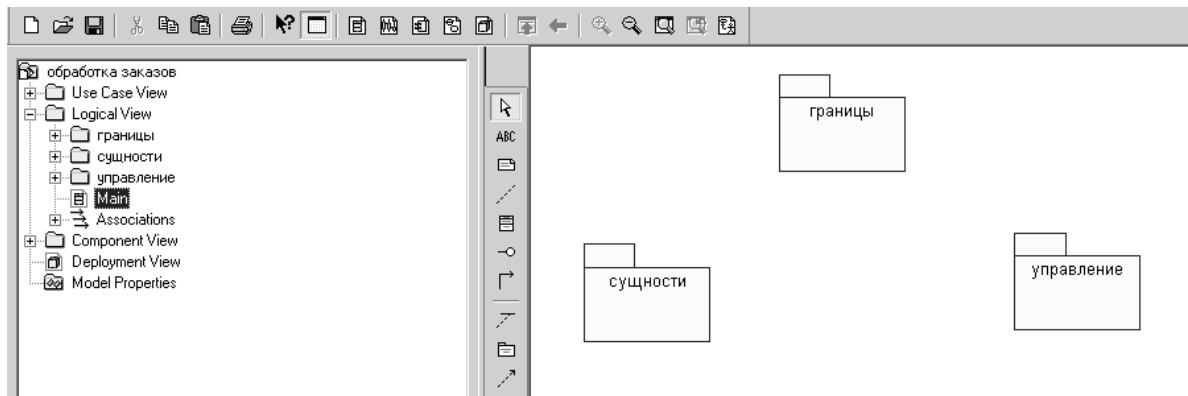


Рис. 7 Главная диаграмма классов в логическом представлении браузера.

Создание диаграммы Классов для сценария "Ввести новый заказ" с отображением всех классов

1. Щелкните правой кнопкой мыши на Логическом представлении браузера.
2. В открывшемся меню выберите пункт New > Class Diagram (Создать > Диаграмма Классов).
3. Назовите новую диаграмму Классов: Ввод нового заказа.
4. Дважды щелкнув мышью на этой диаграмме в браузере, откройте ее.
5. Перетащите из браузера все классы (Выбор_заказа, Заказ_деталей, упр_заказами, Заказ, Упр_транзакциями).

Объединение классов в пакеты

1. В браузере перетащите класс выбор_заказа на пакет Границы.
2. Перетащите класс заказ_деталей на пакет Границы.
3. Перетащите классы Упр_заказами и Упр-транзакциями на пакет Управление.
4. Перетащите класс Заказ на пакет Сущности.

Классы и пакеты в браузере показаны на рис. 9

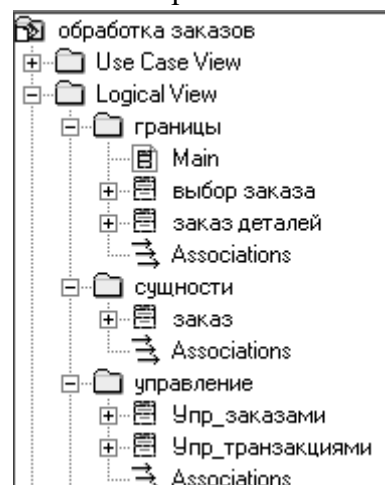


Рис. 8 Представление пакетов и классов

Добавление диаграмм Классов к каждому пакету

1. В браузере щелкните правой кнопкой мыши на пакете Границы.
2. В открывшемся меню выберите пункт New > Class Diagram (Создать > Диаграмма Классов).
3. Введите имя новой диаграммы — Main (Главная).
4. Дважды щелкнув мышью на этой диаграмме, откройте ее.
5. Перетащите на нее из браузера классы выбор_заказа и заказ_деталей.
6. Закройте диаграмму.
- В браузере щелкните правой кнопкой мыши на пакете Сущности.
8. В открывшемся меню выберите пункт New > Class Diagram (Создать > Диаграмма Классов).
9. Введите имя новой диаграммы — Main (Главная).
10. Дважды щелкнув мышью на этой диаграмме, откройте ее.
11. Перетащите на нее из браузера класс Заказ.
12. Закройте диаграмму
13. В браузере щелкните правой кнопкой мыши на пакете Управление
14. В открывшемся меню выберите пункт New > Class Diagram (Создать > Диаграмма Классов).
15. Введите имя новой диаграммы — Main (Главная).
16. Дважды щелкнув мышью на этой диаграмме, откройте ее.
17. Перетащите на нее из браузера классы Упр_заказами и Упр_транзакциями
18. Закройте диаграмму

Задание 4. Уточнение методов и свойств классов.

В этом упражнении к описаниям операций будут добавлены детали, включая параметры и типы возвращаемых значений, и определены атрибуты классов

Постановка проблемы

Для определения атрибутов классов был проанализирован поток событий. В результате к классу Заказ диаграммы Классов были добавлены атрибуты Номер заказа и Имя клиента. Так как в одном заказе можно указать большое количество товаров и у каждого из них имеются свои собственные данные и поведение, было решено моделировать товары как самостоятельные классы, а не как атрибуты класса Заказ.

Добавление атрибутов и операций

Добавим атрибуты и операции к классам диаграммы Классов "Ввод нового заказа".

При этом используем специфические для языка особенности. Установим параметры так, чтобы показывать все атрибуты, все операции и их сигнатуры. Применим нотацию UML.

Этапы выполнения

Настройка

1. В меню модели выберите пункт Tools > Options (Инструменты > Параметры).
2. Перейдите на вкладку Diagram.
3. Убедитесь, что флажок Show visibility (Показать видимость) установлен.
4. Убедитесь, что флажок Show stereotypes (Показать стереотипы) установлен.

5. Убедитесь, что флажок Show operation signatures (Показать сигнатуры операций) установлен.
 6. Убедитесь, что флажки Show all attributes (Показать все атрибуты) и Show all operations (Показать все операции) установлены.
 7. Убедитесь, что флажки Suppress attributes (Подавить атрибуты) и Suppress operations (Подавить операции) сброшены.
 8. Перейдите на вкладку Notation (Нотация).
 9. Убедитесь, что флажок Visibility as icons (Отображать пиктограммы) сброшен.
- Добавление нового класса
1. Найдите в браузере диаграмму Классов варианта использования "Ввести новый заказ".

2. Дважды щелкнув мышью на диаграмме, откройте ее.
3. Нажмите кнопку Class панели инструментов.
4. Щелкните мышью внутри диаграммы, чтобы поместить туда новый класс.
5. Назовите его Позич_заказа.
6. Назначьте этому классу стереотип Entity.
7. В браузере перетащите класс в пакет Сущности.

Добавление атрибутов

1. Щелкните правой кнопкой мыши на классе Заказ.
2. В открывшемся меню выберите пункт New Attribute (Создать атрибут).
3. Введите новый атрибут:
OrderNumber : Integer
4. Нажмите клавишу Enter
5. Введите следующий атрибут:
CustomerName : String.
6. Повторив шаги 4 и 5, добавьте атрибуты:

OrderDate : Date

OrderFillDate : Date

Если тип атрибута не появляется в выпадающем списке, то введите его от руки и он далее будет появляться.

7. Щелкните правой кнопкой мыши на классе Позич_заказа.
8. В открывшемся меню выберите пункт New Attribute (Создать атрибут).
9. Введите новый атрибут:

ItemID : Integer.

10. Нажмите клавишу Enter.

11. Введите следующий атрибут:

ItemDescription : String.

Добавление операций к классу Позич_заказа

1. Щелкните правой кнопкой мыши на классе Позич_заказа.
2. В открывшемся меню выберите пункт New Operation (Создать операцию).
3. Введите новую операцию:
Создать()

4. Нажмите клавишу Enter.

5. Введите следующую операцию:

Взять_информацию()

6. Нажмите клавишу Enter.

7. Введите операцию:

Дать_информацию()

Подробное описание операций с помощью диаграммы Классов

1. Щелкнув мышью на классе Заказ, выделите его.

2. Щелкните на этом классе еще раз, чтобы переместить курсор внутрь.

3. Отредактируйте операцию Создать(), чтобы она выглядела следующим образом:

Создать() : Boolean

4. Отредактируйте операцию Взять_информацию:

Взять_информацию (OrderNum : Integer, Customer : String, OrderDate : Date, FillDate : Date): Boolean

5. Отредактируйте операцию Дать_информацию;

Дать_информацию(): String

Подробное описание операций с помощью браузера

1. Найдите в браузере класс Позиц_заказа.

2. Раскройте этот класс, щелкнув на значке "+" рядом с ним. В браузере появятся атрибуты и операции класса.

3. Дважды щелкнув мышью на операции Дать_информацию() , откройте окно ее спецификации:

4. В раскрывающемся списке Return class (Возвращаемый класс) укажите String.

5. Щелкнув мышью на кнопке ОК, закройте окно спецификации операции.

6. Дважды щелкните в браузере на операции Дать_информацию () класса Позиц_заказа, чтобы открыть окно ее спецификации.

7. В раскрывающемся списке Return class укажите Boolean.

8. Перейдите на вкладку Detail(Подробно).

9. Щелкните правой кнопкой мыши в области аргументов, чтобы добавить туда новый параметр:

10. В открывшемся меню выберите пункт Insert (Вставить). Rose добавит аргумент под названием argname.

11. Щелкнув один раз на этом слове, выделите его и измените имя аргумента на ID.

12. Щелкните на колонке Type (Тип). В раскрывающемся списке типов выберите Integer (Если этого либо иного необходимого типа не будет- введите его вручную).

13. Щелкните на колонке Default (По умолчанию), чтобы добавить значение аргумента по умолчанию. Введите число 0.

14. Нажав на кнопку ОК, закройте окно спецификации операции.

15. Дважды щелкните на операции Создать() класса Позиц_заказа, чтобы открыть окно ее спецификации.

16. В раскрывающемся списке Return class укажите Boolean.

17. Нажав на кнопку ОК, закройте окно спецификации операции.

Подробное описание операций

1. Используя браузер или диаграмму Классов, введите следующие сигнатуры операций класса Заказ_деталей:

Открыть() : Boolean

Сохранить заказ() : Boolean

2. Используя браузер или диаграмму Классов, введите сигнатуру операций класса

Выбор_заказа:

Создать() : Boolean

3. Используя браузер или диаграмму Классов, введите сигнатуру операций класса

Упр_заказами:

Сохранить заказ(OrderID : Integer) : Boolean

4. Используя браузер или диаграмму Классов, введите сигнатуры операций класса

Упр_транзакциями:

Сохранить заказ(OrderID : Integer) : Boolean

Сохранить информацию() : Integer

Задание 5. Описание связей между классами

В этом упражнении определяются связи между классами, участвующими в варианте использования "Ввести новый заказ".

Постановка задачи

Чтобы найти связи, были просмотрены диаграммы Последовательности. Все взаимодействующие там классы нуждались в определении соответствующих связей на диаграммах Классов. После обнаружения связи были добавлены на диаграммы классов.

Добавление связей

Добавим связи к классам, принимающим участие в варианте использования "Ввести новый заказ".

Этапы выполнения упражнения

Настройка

1. Найдите в браузере диаграмму Классов "Ввод нового заказа",

2. Дважды щелкнув на диаграмме, откройте ее.

3. Проверьте, имеется ли в панели инструментов диаграммы кнопка Unidirectional Association (Однонаправленная ассоциация). Если ее нет, продолжите настройку, выполнив шаги 4 и 5. Если есть, приступайте к выполнению самого упражнения.

4. Щелкните правой кнопкой мыши на панели инструментов диаграммы и в открывшемся меню выберите пункт Customize(Настроить),

5. Добавьте на панель кнопку Creates A Unidirectional Association (Создать однонаправленную ассоциацию).

Добавление ассоциаций

1. Нажмите кнопку Unidirectional Association панели инструментов.

2. Проведите ассоциацию от класса выбор_заказа к классу заказ_деталей.

3. Повторите шаги 1 и 2, создав ассоциации:

- От класса заказ_деталей к классу упр_заказами
- От класса упр_заказами к классу Заказ

- От класса упр_заказами к классу упр_транзакциями
- От класса упр_транзакциями к классу Заказ
- От класса упр_транзакциями к классу Позиц_заказа
- От класса Заказ к классу Позиц_заказа

4. Щелкните правой кнопкой мыши на однонаправленной ассоциации между классами выбор_заказа и заказ_деталей класса выбор_заказа.

5. В открывшемся меню выберите пункт Multiplicity > Zero or One (Множественность >- Нуль или один),

6. Щелкните правой кнопкой мыши на другом конце однонаправленной ассоциации.

7. В открывшемся меню выберите пункт Multiplicity > Zero or One (Множественность > Нуль или один),

8. Повторите шаги 4—7, добавив на диаграмму значения множественности для остальных ассоциаций, как показано на рис. 10

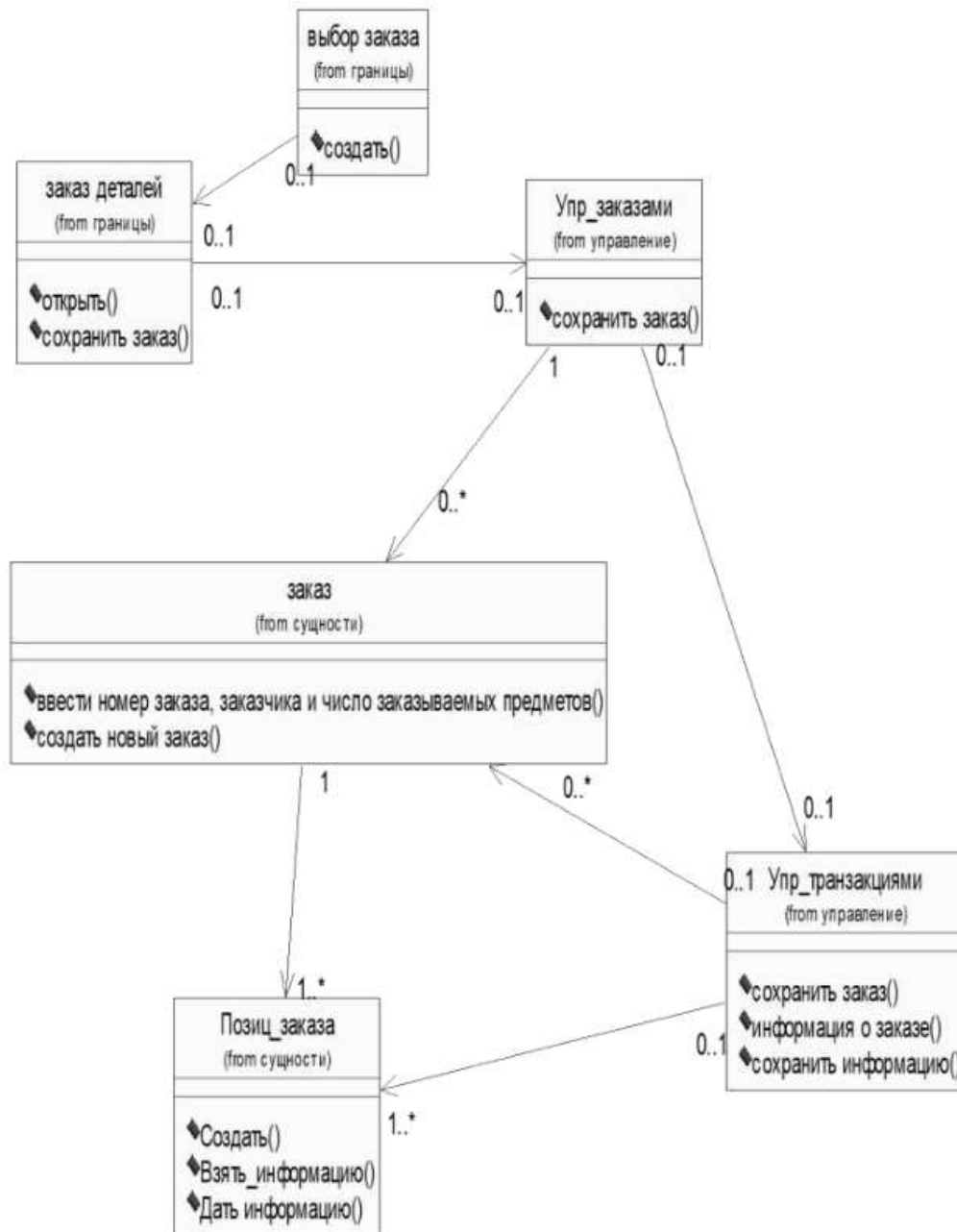


Рис. 10 Ассоциации сценария "Ввести новый заказ"

Задание 6. Исключение кириллизованного текста в информации классов.

Разработанные ранее модели, предназначенные для описания предметной области используют кириллизованную символику, недопустимую для большинства языков программирования. Выполните замену русского текста на латинский шрифт. Для этой цели сохраните предыдущую модель под другим именем и далее работайте с новым файлом (что бы при необходимости можно было бы вернуться к бизнес- процессам, описанным русским шрифтом).

Этапы выполнения

Этап 1. Используя меню (Файл-> Сохранить как) сохраните данную модель под другим именем (например Заказ1) в той же папке, что и исходная модель.

Работайте далее с копией модели (то есть Заказ1).

Этап 2. Переименуйте классы и их спецификации таким образом, чтобы использовался только латинский шрифт. Замените имя класса

Заказ_деталей на OrderDetail

Выбор_заказа на OrderOptions

Заказ на Order

Упр_заказами на OrderMgr

Позиц_заказа на OrderItem

Упр_транзакциями на TransactionMgr

Измените имена операций таким образом, чтобы рис.10 преобразовался в рис. 11. Для этого, измените операцию класса OrderOptions

Открыть() на Open()

Класса OrderDetail

Открыть() на Open()

Сохранить заказ() на Save()

Класса Order

Ввести номер заказа, заказчика и число заказываемых предметов() на SetInfo()

Сохранить_заказ() на Save()

Класса OrderMgr

Сохранить заказ() на SaveOrder()

Класса TransactionMgr

Сохранить заказ() на SaveOrder()

Сохранить информацию о заказе() на Commit()

Создать_заказ() на SubmitInfo()

Класса OrderItem

Создать() на Create()

Взять_информацию() на GetInfo()

Дать_информацию на SetInfo()

Переименуйте имена пакетов

Границы на Boundaries

Сущности на Entity

Контроль на Control

Добавление стереотипов к классам

1. Щелкните правой кнопкой мыши на классе OrderOptions диаграммы.
2. В открывшемся меню выберите пункт Open Specification (Открыть спецификацию).
3. В поле стереотипа выберите из выпадающего списка слово Boundary (если его нет, то введите).
4. Нажмите на кнопку ОК.

5. Повторив шаги 1—4, свяжите классы OrderDetail со стереотипом Boundary, OrderMgr и TransactionMgr со стереотипом Control, а класс Order и OderItem— со стереотипом Entity.

Теперь диаграмма Классов должна иметь вид, показанный на рис. 11.

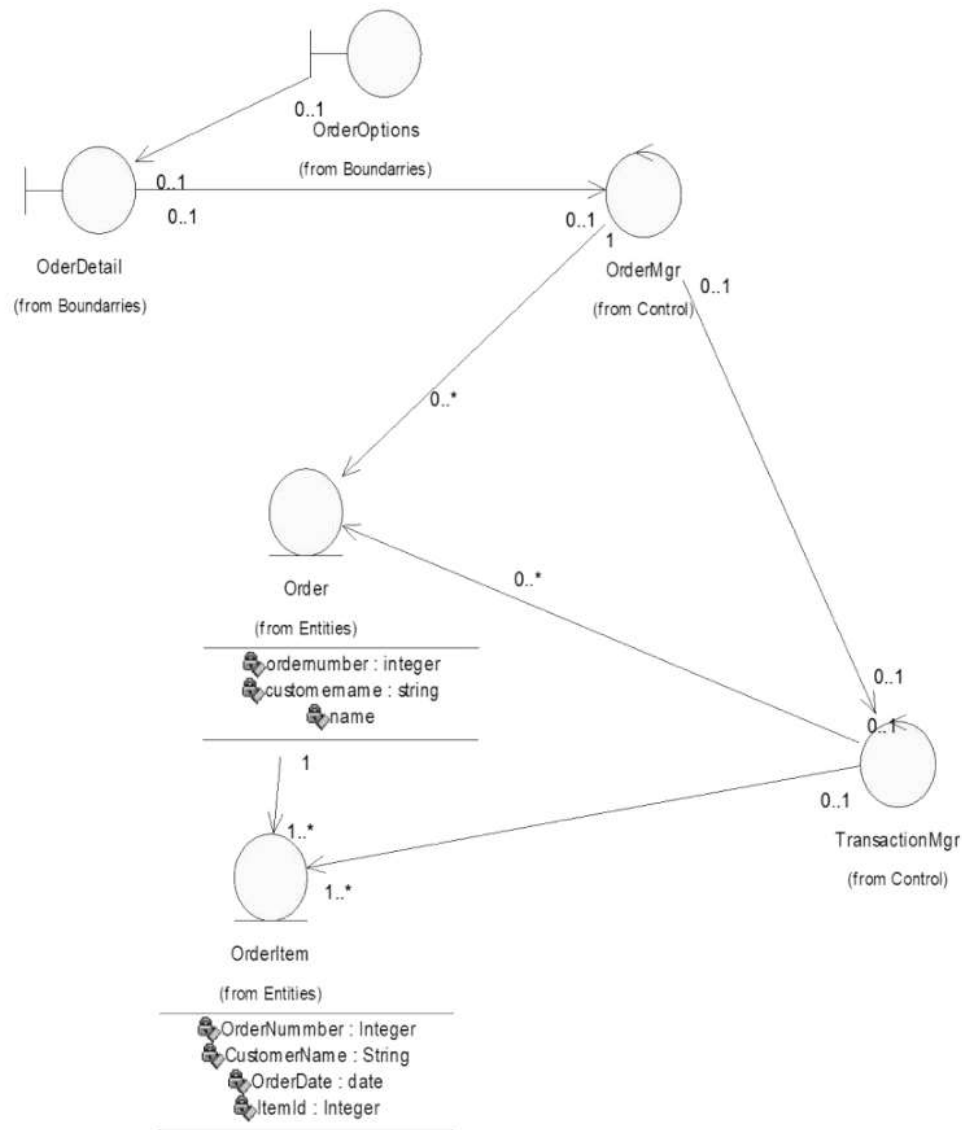


Рис. 11 Основная диаграмма классов

Замечание. На диаграмме рис. 11 возможно визуальное представление классов не в виде иконок, а в виде дополнительной строки текста с именем стереотипа. За этот вид отвечает метка установленная либо на icon либо на label (Class> Open Specification> Options> Label)

Лабораторная работа «Построение диаграммы компонентов»

Цель работы: исследование процесса построения диаграммы компонентов и в заданной предметной области с помощью пакета Rational Rose

Задание 1. Построение диаграммы Компонентов

Этапы выполнения

Так как эта модель связана с конкретным языком программирования, то в настройках это необходимо отметить. Выполнить Tools>Options>Notations>Default Language и из выпадающего списка языков программирования выбрать Delphi.

Создание пакетов компонентов

1. Щелкните правой кнопкой мыши на представлении компонентов в браузере.
2. В открывшемся меню выберите пункт New > Package (Создать > Пакет}.
3. Назовите пакет Entities (Сущности).
4. Повторив шаги с первого по третий, создайте пакеты Boundaries (Границы) и Control (Управление).

Добавление пакетов на Главную диаграмму Компонентов

1. Откройте Главную диаграмму Компонентов, дважды щелкнув на ней мышью,
2. Перетащите пакеты Entities, Boundary и Control из браузера на Главную диаграмму.

Отображение зависимостей между пакетами

1. Нажмите кнопку Dependency (Зависимость) панели инструментов.
2. Щелкните мышью на пакете Boundary Главной диаграммы Компонентов.
3. Проведите линию зависимости к пакету Control.
4. Повторив шаги 1 — 3, проведите зависимость от пакета Control к пакету Entities.

В результате диаграмма примет вид рис. 12

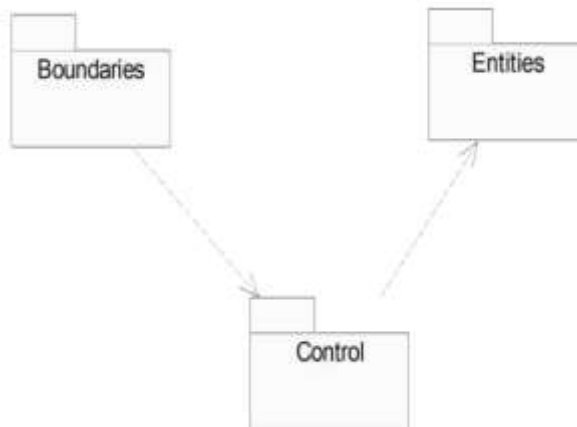


Рис. 12 Зависимости между пакетами

Добавление компонентов к пакетам и отображение зависимостей

1. Дважды щелкнув мышью на пакете Entities Главной диаграммы Компонентов, откройте Главную диаграмму Компонентов этого пакета.
2. Нажмите кнопку Package Specification (Спецификация пакета) панели инструментов.
3. Поместите спецификацию пакета на диаграмму.
4. Введите имя спецификации пакета — OrderItem_.
5. Повторив шаги 2—4, добавьте спецификацию пакета Order_.
6. Нажмите кнопку Dependency (Зависимость) панели инструментов.
7. Щелкните мышью на спецификации пакета OrderItem_.
8. Проведите линию зависимости к спецификации пакета OrderItem_.
9. С помощью описанного метода создайте следующие компоненты и зависимости:
Для пакета Boundaries:
 - Спецификацию пакета Orderoptions_

- Спецификацию пакета OrderDetail_

Зависимости в пакете Boundaries:

- От спецификации пакета Orderoptions_ к спецификации пакета OrderDetail_

Для пакета Control:

- Спецификацию пакета OrderMgr_
- Спецификацию пакета TransactionMgr_

Зависимости в пакете Control:

- От спецификации пакета OrderMgr_ к спецификации пакета TransactionMgr_

Создание диаграммы Компонентов системы

1. Щелкните правой кнопкой мыши на представлении Компонентов в браузере.
2. В открывшемся меню выберите пункт New > Component Diagram (Создать > Диаграмма Компонентов).

3. Назовите новую диаграмму System.

4. Дважды щелкните на этой диаграмме мышью.

Размещение компонентов на диаграмме Компонентов системы

1. Разверните в браузере пакет компонентов Entities, чтобы открыть его.
2. Щелкните мышью на спецификации пакета Order_ в пакете компонентов Entities.
3. Перетащите эту спецификацию на диаграмму.
4. Повторив шаги 2 и 3, поместите на диаграмму спецификацию пакета OrderItem_.
5. С помощью этого метода поместите на диаграмму следующие компоненты:

Из пакета компонентов Boundaries:

- Спецификацию пакета Orderoptions_
- Спецификацию пакета OrderDetail_

Из пакета компонентов Control:

- Спецификацию пакета OrderMgr_
- Спецификацию пакета TransactionMgr_

6. Нажмите кнопку Task Specification (Спецификация задачи) панели инструментов.

7. Поместите на диаграмму спецификацию задачи и назовите ее OrderClientExe.

8. Повторите шаги 6 и 7 для спецификации задачи OrderServerExe.

Добавление оставшихся зависимостей на диаграмму Компонентов системы

Уже существующие зависимости будут автоматически показаны на диаграмме Компонентов системы после добавления туда соответствующих компонентов. Теперь нужно добавить остальные зависимости.

1. Нажмите кнопку Dependency (Зависимость) панели инструментов.

2. Щелкните мышью на спецификации пакета OrderDetail_

3. Проведите линию зависимости к спецификации пакета OrderDetail_

4. Повторив шаги 1 — 3, создайте следующие зависимости:

- От спецификации пакета OrderMgr_ к спецификации пакета Order_
- От спецификации пакета TransactionMgr_

к спецификации пакета OrderItem_

- От спецификации пакета TransactionMgr_

к спецификации пакета Order_

- От спецификации задачи OrderClientExe к спецификации пакета Orderoptions_

От спецификации задачи OrderServerExe к спецификации пакета OrderMgr_r

Соотнесение классов с компонентами

1. В Логическом представлении браузера найдите класс Order пакета Entities.

2. Перетащите этот класс на спецификацию пакета компонента Order_ в представлении Компонентов браузера, В результате класс Order будет соотнесен со спецификацией пакета компонента Order_.

3. Повторив шаги 1 — 2, соотнесите классы со следующими компонентами:

- Класс OrderItem со спецификацией пакета OrderItem_

- Класс Orderoptions со спецификацией пакета Orderoptions_

- Класс OrderDetail со спецификацией пакета OrderDetail_1

- Класс OrderMgr со спецификацией пакета OrderMgr_

- Класс TransactionMgr со спецификацией пакета TransactionMgr_

В результате в браузере после имени класса, в скобках появятся имена компонентов, с которыми этот класс связан (рис. 13)

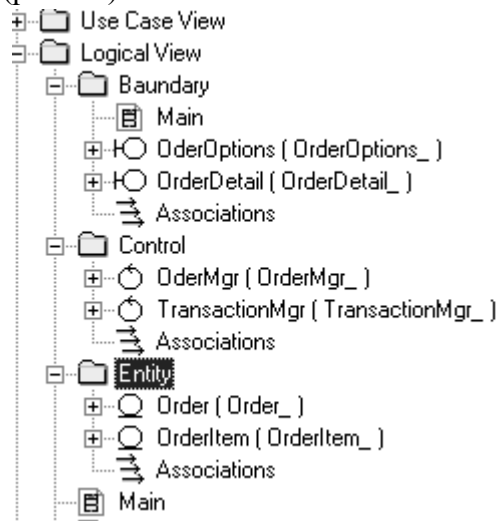


Рис. 13 Представление компонентов и классов в браузере

Лабораторная работа «Построение диаграмм потоков данных»

Цель работы: исследование процесса построения диаграммы потоков данных и диаграммы размещения в заданной предметной области с помощью пакета Rational Rose и Delphi

Задание 1. Кодогенерация проекта в Delphi.

Теперь вся информация подготовлена к тому, чтобы запрограммировать классы с их методами и операциями.

Для выполнения кодогенерации в среде Delphi необходимо выполнить следующую последовательность действий:

- протестировать модель на логические непротиворечия;
- настроить (или проверить настройки) среду на законы кодогенерации (соответствие элемента модели Rose элементу кода Delphi);
- создать имя проекта Delphi и выполнить кодогенерацию.

Этапы выполнения упражнения.

1) Протестируйте модель Tools->Ceck Model. Просмотрите log файл на наличие ошибок. Если файл не виден- выполните команду file->Save Log As и введите имя файла (по умолчанию error.log). Затем его просмотрите и, при необходимости, исправьте ошибки. К наиболее распространенным ошибкам относятся такие, как неотображение сообщений на операции или несоотнесение объектов с классами на диаграммах взаимодействия. С помощью пункта меню Check Model можно выявить большую часть неточностей и ошибок в модели.

2) Пункт меню Access Violations позволяет обнаружить нарушения правил доступа, возникающие тогда, когда существует связь между двумя классами разных пакетов. При этом связи между самими пакетами нет. Например, если существует связь между классами Order пакета Entities и OrderManager пакета Control, то обязательно должна существовать и связь между пакетами Entities и Control. Если последняя связь не установлена, Rose выявит нарушение правил доступа. Чтобы обнаружить нарушение правил доступа:

Выберите в меню Report > Show Access Violations.

Проанализируйте все нарушения правил доступа.

3) Выполните Tools>Options>Notation>Default Language и из выпадающего списка выберите язык программирования Delphi.

4) Проверьте правильность установок кодогенерации по умолчанию (default). Для этого выполните Tools->Options->Delphi и последовательно переберите из выпадающего списка поля Type все элементы. Сравните установки в поле Model Propities с данными (default) из таблицы Приложения А. В случае несоответствия- исправьте.

5) Выполните Tools> Ensemble Tools>Rose Delphi Link (рис.14)

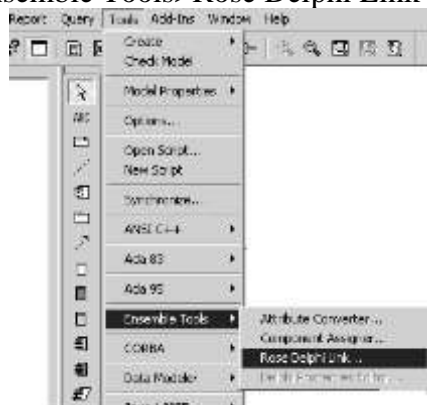


Рис. 14. Меню для выбора процесса кодогенерации

В результате появится соответствующая экранная форма. Выполните на этой форме File>New Proect. Появится форма с браузером. Введите имя файла и место на диске, куда будет сохранено имя сгенерированного проекта в Delphi. Например, NewProect.dpr и нажмите Открыть. В результате форма примет вид (рис. 15)

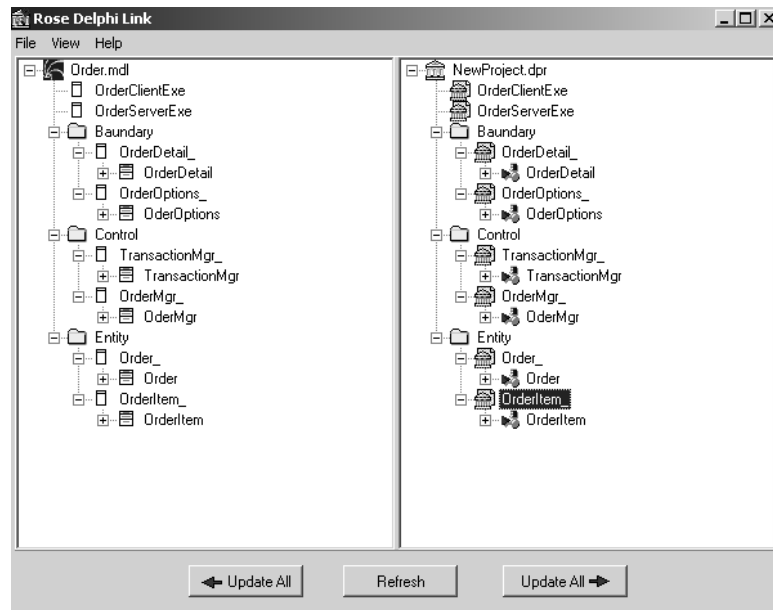


Рис. 15. Представление результатов кодогенерации в окне Rose Delphi Link

5. Через проводник Windows найдите папку проекта Delphi. С помощью программы Блокнот просмотрите содержимое всех файлов. В приложении В к руководству приведено содержимое всех файлов проекта.

Задание 2. Анализ Delphi проекта, добавление визуальных объектов, реинжиниринг в Rose

1) Запустите на выполнение программу Delphi и загрузите сгенерированный проект (Proect1.dpr). Проверьте, что проект содержит все модули и присмотрите их содержимое через редактор Delphi.

2) Создайте в проекте Delphi новую форму с Name Form1. Поместите на форму компонент MainMenu (главное меню)

3) С помощью Menu Designer введите две позиции горизонтального меню с названиями (полями Caption) Oder и OderItem.

4) Для Oder введите две строки вертикального меню с Caption Create и SubmitInfo. Для OderItem введите одну строку вертикального меню с названием GetInfo.

5) Сохраните проект в Delphi.

Реинжиниринг Delphi проекта в модель Rose

1) Вернитесь в проект Rose и откройте окно проектов Rose Delphi Link. Проверьте, что открыт именно тот проект, для которого выполнялась кодогенерация.

2) Курсором мыши нажмите клавишу Update ALL со стрелкой влево (обновление модели Rose на основе изменений проекта Delphi). В результате в модели Rose должны произойти определенные изменения (рис. 16):

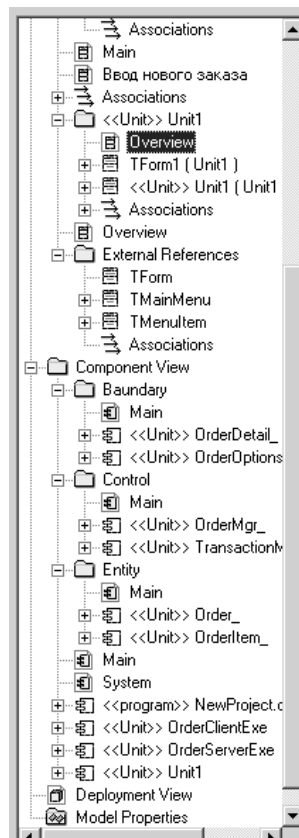


Рис. 16. Окно Rose Delphi Link после кодогенерации

- в представлении Logic View создан новый пакет Unit1 и External References (Внешние ссылки). Внутри второго пакета созданы три класса TForm, TMainMenu и TMenuItem, которые использовались при развитии проекта Delphi. Отметим, что эта папка не создавалась бы, если бы мы при первоначальном создании проекта включили в него пакет классов Delphi FreeimWork.

- в этом же представлении в пакете Unit1 создан класс TForm1 и Unit1 оба соотнесенные с вновь созданным компонентом Unit1. Кроме того, в этом же пакете создавалась диаграмма классов Overview, содержимое которой показано на рис.17

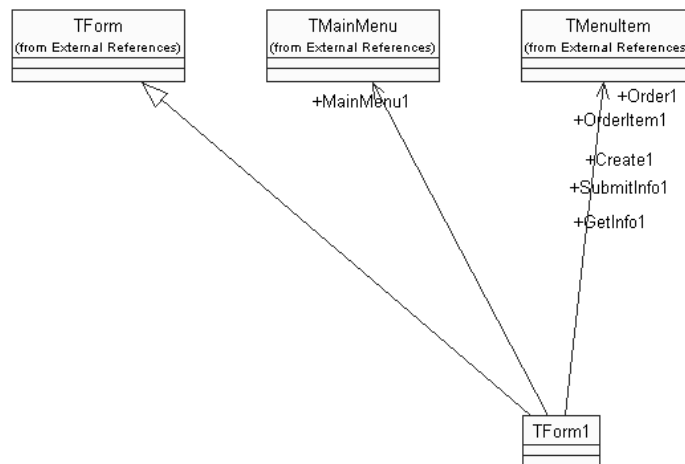


Рис. 17. Результаты реинжиниринга проекта Delphi в Rose

Задание 3. Построение диаграммы размещения

В этом задании создается диаграмма Размещения для системы обработки заказов.

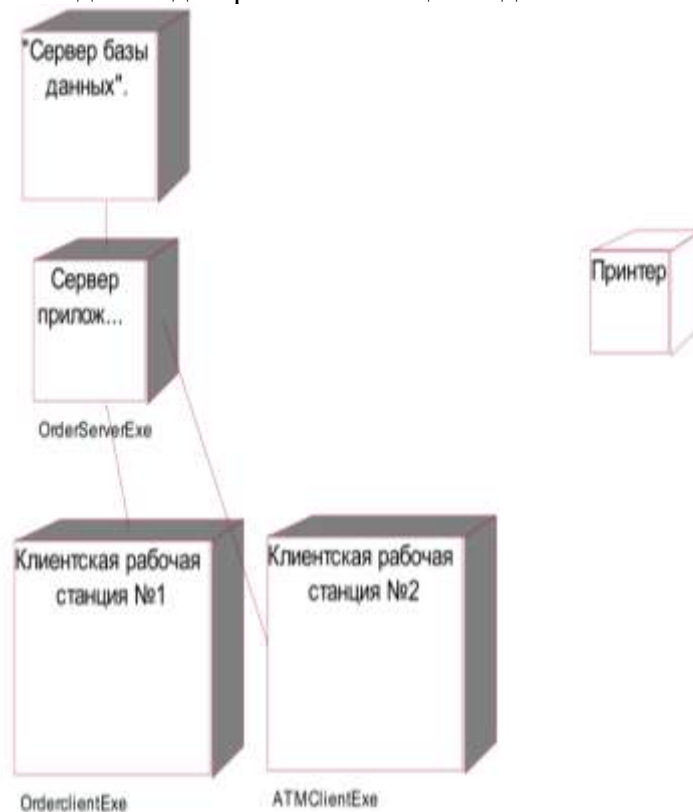


Рис. 18 Диаграмма размещения для модельной задачи

Этапы выполнения

Добавление узлов к диаграмме Размещения

1. Дважды щелкнув мышью на представлении Размещения в браузере, откройте диаграмму Размещения.

2. Нажмите кнопку Processor (Процессор) панели инструментов.

3. Щелкнув мышью на диаграмме, поместите туда процессор.

4. Введите имя процессора "Сервер базы данных".

5. Повторив шаги 2—4, добавьте следующие процессоры:

-Сервер приложения

- Клиентская рабочая станция №1

- Клиентская рабочая станция №2

6. На панели инструментов нажмите кнопку Devices (Устройство).

7. Щелкнув мышью на диаграмме, поместите туда устройство.

8. Назовите его "Принтер".

Добавление связей

1. Нажмите кнопку Connection (Связь) панели инструментов.

2. Щелкните мышью на процессоре "Сервер базы данных".

3. Проведите линию связи к процессору "Сервер приложения".

4. Повторив шаги 1 — 3, добавьте следующие связи;

- От процессора "Сервер приложения" к процессору "Клиентская рабочая станция №1"

- От процессора "Сервер приложения" к процессору "Клиентская рабочая станция №2"

- От процессора "Сервер приложения" к устройству "Принтер"

Добавление процессов

1. Щелкните правой кнопкой мыши на процессоре "Сервер приложения" в браузере.

2. В открывшемся меню выберите пункт New > Process (Создать > Процесс),
3. Введите имя процесса — OrderServerExe.
4. Повторив шаги 1 — 3, добавьте процессы:
 - Процесс OrderclientExe на процессоре "Клиентская рабочая станция №1"
 - Процесс ATMClientExe на процессоре "Клиентская рабочая станция №2"

Показ процессов на диаграмме

1. Щелкните правой кнопкой мыши на процессоре "Сервер приложения".
2. В открывшемся меню выберите пункт Show Process (Показать процессы).
3. Повторив шаги 1 и 2, покажите процессы на следующих процессорах:
 - Клиентская рабочая станция №1
 - Клиентская рабочая станция №2

Задания для проведения дифференцированного зачета МДК 02.01

ЗАДАНИЕ № 1.2

Вам поручили разработать дизайн базы данных, а также диаграмму прецедентов для фитнес центра «F.I.T.». Вам не нужно разрабатывать систему, вам необходимо:

- создать ERD – диаграмму и диаграмму прецедентов;
- используя специализированные графические средства построить архитектуру программного продукта.

Вводные данные к предполагаемой базе данных

Организаторы Marathon Skills купили сеть фитнес-центров. Вопрос: «F.I.T.». к сожалению, степень автоматизации существующих оставляет желать лучшего, поэтому было принято решение разработать информационную систему для автоматизации процессов. Вопрос: Поручить вам проектирование будущей системы

Примечание: Вы НЕ создаете базу данных. Это не требуется. Вы должны разработать ERD словарь данных, которые определяют базу данных, а также построить диаграмму прецедентов.

Описание деятельности сети фитнес - клубов «F.I.T.» и текущих бизнес-процессов Сеть фитнес - клубов «F.I.T.» представляет членам клуба полный комплекс фитнес- и wellness программ, групповой и индивидуальный тренинг, тренажерные залы, косметические процедуры, фитнес - бары и солярии.

Фитнес-центр предоставляет свои услуги клиентам всех возрастов и уровней подготовки. Клиентами центра являются люди среднего и высокого достатка, ведущие активный образ жизни.

На сегодняшний день в «F.I.T.» занимается более 1 500 человек.

На входе в клуб клиента встречает администратор и определяет его потребности. Клиент может обратиться только в то отделение клуба, в котором он планирует получить услугу.

Если клиент хотел бы получить косметические услуги, то администратор направляет его в косметический кабинет. Специалист косметического кабинета помогает клиенту определиться с перечнем услуг, информирует об их стоимости и времени проведения. Клиент оплачивает администратору стоимость услуг и получает квитанцию об оплате. Администратор заносит в тетрадь ФИО клиента, название услуги, ее стоимость. На основании квитанции клиенту предоставляется услуга в косметическом кабинете в установленное время.

Если клиент планирует заняться фитнесом, то администратор информирует его о фитнес - программах, расписании (Приложение 1.1), условиях покупки клубной карты или абонемента (Приложение 1.2) и его использования.

Клиент оплачивает фитнес услуги, администратор выдает ему абонемент, клубную карту или сертификат на предоставление дополнительных услуг. Организация может заключить договор на обслуживание своих сотрудников (Приложение 1.3).

Клиент обязан записываться на тренировки, чтобы администратор смог контролировать количество человек в группе. При записи на тренировку администратор вносит ФИО клиента в специальный лист записи (Приложение 1.4). При посещении клиентами тренировок ведется фактический учет. Администратор обязан согласовывать изменения в расписании с тренерским составом клуба и информировать клиентов об изменениях.

Постановка задачи

Для оценки трудозатрат на построение информационной системы для сети фитнес-клубов необходимо разработать:

1. Логическую модель данных, включающую:
 - Определение хранимых сущностей;
 - Определение связей между сущностями, их кратностей; ☐ Определение доменов для хранимых значений (типов данных).
2. Диаграмму прецедентов, включающую:
 - Определение актеров;
 - Определение вариантов использования системы;
 - Определение отношений между актерами и прецедентами.

Условия выполнения задания:

1. Место (время) выполнения задания: компьютерная лаборатория
2. Технические средства обучения: ПК
3. Задание состоит из практической части (решения задачи)
4. Максимальное время выполнения задания: 45 мин

Критерии оценки:

Sub Criteria	Aspect - Description	Extra Aspect Description	Max Mark
Дизайн базы данных (ERD)	Все объекты рассмотрены и определены	Минус 0,4 балла за каждый отсутствующий / неправильный объект	2,00
	Все атрибуты (поля) рассмотрены и определены	Минус 0,2 балла за каждый отсутствующий / неправильный атрибут	2,00
	Отношения определены правильно	Минус 0,4 балла за каждое отсутствующее / неправильное отношение	2,00
Диаграмма прецедентов	Все актеры определены и перечислены	Минус 0,5 балла за каждого отсутствующего / неправильного актера	2,00
	Определены все варианты использования системы	Минус 0,4 балла за каждый неправильный/отсутствующий вариант использования	3,50
	Определены отношения между актерами и прецедентами	Минус 0,3 балла за каждое отсутствующее / неправильное отношение	3,00
Итого			14,5

Оценка 5	Оценка 4	Оценка 3	Оценка 2
----------	----------	----------	----------

14,5-11,6	11,5-9,0	8,9-5,0	Менее 4,9
-----------	----------	---------	-----------

Приложение 1.1 Расписание

занятий центрального отделения фитнес-клуба

Фитнес зал							
	Понедель- ник	Вторник	Среда	Четверг	Пятни- ца	Суббота	Воскресе- нье
10: 00 – 11: 00		Интер- вальная трени- ровка МАРИНА		Problem zone МАРИ- НА			
11: 00 – 12: 00							Футбол АЛЛА
12: 00 – 13: 00							Танце- вальная аэробика + кор. фи- гуры АЛЛА
13: 00 – 14: 00		Problem zone МАРИНА		Поху- дейка МАРИ- НА		Худеем танцуя! МАРИНА	
15: 00 – 16: 00							
16: 00 – 17: 00		Детская аэробика ИРИНА (8-16 лет)		Детская аэроби- ка ИРИНА (8-16 лет)		Интер- вальная трениров- ка ЛЕНА	BODY COMBAT + кор. фи- гуры АНТОН
17: 00 – 18: 00	Step + Power Class КСЮША	Коррек- ция фигу- ры	Интер- вальная трени-	Поху- дейка ВИКА	Коррек- ция фигу-	Functional cuts ЛЕНА	Интер- вальная трениров- ка КСЮША
		ВИКА	ровка КСЮША		ры КСЮША		

18:00 – 19:00	Похудейка НАДЯ	Problem zone КСЮША	Function al cuts НАДЯ	Кор-рекция фигуры КСЮ-ША	Интер-вальная трениров-ка НАДЯ		Problem zone КСЮША
19:00 – 20:00	Интер-вальная трениров-ка НАДЯ	Интер-вальная трениров-ка КСЮША	Фитнесйо-га <i>(для начи-нающих и среднего уровня подготовки)</i> НАДЯ	Problem zone КСЮ-ША	Proble m zone НАДЯ		
20:00 – 21:00	BODY COMBAT + кор. фигу-ры АНТОН	Тай-бо + кор. фигу-ры НАТАЛЬЯ	BODY COMBAT + кор. фигуры АНТОН	Тай-бо + кор. фигуры НАТА-ЛЬЯ	BODY COMBAT + кор. фигуры АНТОН		

Хореографический зал							
	Понедельник	Втор-ник	Среда	Чет-верг	Пятница	Суббота	Воскресе-нье
16:00 – 17:00						Стриппла-стика + Stretch ОКСАНА	Аэробика в стиле РЕТРО ИРИНА
17:45 – 18:45		Pilates mini Ball ОЛЯ	Аэробика в стиле РЕ-ТРО ИРИНА	Pilate s mini Ball ОЛЯ	Power-йога ЛЕНА Красильни-кова		
18:45 – 19:45	Стриппла-стика + Stretch ОКСАНА		Стриппла-стика + Stretch ОКСАНА		Худеем танцуя! МАРИНА		
19:45 – 20:45	Lady's Style МАРИНА		Lady's Style МАРИНА				

Железный зал							
	Понедель-ник	Вторник	Среда	Четверг	Пятни-ца	Суббота	Воскресе-нье

18:1 5 – 19:1 5		Hot Iron I ЕЛЕНА		Hot Iron I ЕЛЕНА		16:00 – 18:00 ЦИ- ГУН АН- ДРЕЙ	
18:4 5 – 19:4 5	Hot Iron I КСЮША		Hot Iron I КСЮ- ША		Hot Iron I КСЮ- ША		
19:1 5 – 20:1 5		Hot Iron I ВИКА		Hot Iron I ВИКА			
19:4 5 – 20:4 5	Hot Iron I МАША		Hot Iron I МАША		Hot Iron I МАША		
20:1 5 – 21:1 5		Стриппла- стика + Stretch		Стриппла- стика + Stretch ОКСАНА			
		ОКСАНА					

Йога - студия							
	Понедельник	Вторник	Среда	Четверг	Пятница	Суббота	Воскресе- нье
09:0 0 – 10:0 0	YOGA ПО- ЛИНА		YOGA ПО- ЛИНА		YOGA ПОЛИ- НА		
11:0 0 – 12:0 0							YOGA ПОЛИНА
12:0 0 – 13:0 0						Pilates Ball ОЛЯ	12:00 – 13:30 YOGA ПОЛИНА
14:0 0 – 15:0 0							Фитнес - йога ТАТЬЯ- НА

17:15 – 18:15		YOGA ПОЛИ-НА Фикс. группа		YOGA ПОЛИ-НА Фикс. группа		17:00 – 18:00 Yoga, Stretch, Relax (для начинающих) ЕВГЕНИЯ	
17:30 – 18:30	Power-йога ЛЕНА Красильникова		Powerйога ЛЕНА Красильникова		Pilates Ball ОЛЯ		Фитнес - йога ОЛЯ
18:15 – 19:15		YOGA ПОЛИ-НА		YOGA ПОЛИ-НА			
18:30 – 19:30	Pilates Mix ОЛЯ		Pilates Ball ОЛЯ		Фитнес - йога ОЛЯ		
19:15 – 20:15		Pilates Mix ОЛЯ		Pilates Mix ОЛЯ			
19:30 – 20:30	YOGA ПОЛИНА Фикс. группа		YOGA ПОЛИНА Фикс. группа				
20:15 – 21:15		Фитнес - йога ТАТЬЯНА		Фитнес - йога ТАТЬЯНА			

Приложение 1.2

Условия оплаты фитнес - программ Клубные карты на неограниченное использование фитнес - программ:

- «Спринт» (1 месяц) - 2 900 руб.
- «Экспресс» (3 месяца) - 6 500 руб.
- «Энерджи» (6 месяцев) - 11 300 руб.
- «Профи» (12 месяцев) - 20 500 руб.

Скидки на клубные карты:

Вид	Размер (%)	Условия
Семейная	5	При одновременной покупке двух клубных карт семейной парой. Предоставляется при предъявлении документа, подтверждающего брак.

	50	При покупке клубной карты ребенку, возрастом до 14 лет, родители которого являются обладателями клубных карт.
--	----	---

Скидки на клубные карты и на абонементы:

Вид	Размер (%)	Условия
Корпоративная	5	При единовременной покупке от 5-ти до 10-ти клубных карт или абонементов.
	10	При единовременной покупке от 10 абонементов и клубных карт.

Абонементы на 1 месяц:

- Фитнес-утро — 1 300 руб.
- Фитнес-вечер — 1 950 руб.

Дополнительные услуги:

- Персональная тренировка в тренажерном зале для обладателей клубной карты – 400 руб.
- Персональная тренировка в тренажерном зале + посещение сауны + прокат полотенца - 600 руб.
- «Комби» - 5 персональных тренировок в тренажерном зале+ посещение финской сауны+ прокат полотенца – 2 800 руб.
- «XL комби» - 10 персональных тренировок в тренажерном зале + посещение финской сауны + прокат полотенца – 5 500 руб.
- Персональная тренировка по фитнес аэробике, йоге - 700 руб.
- Персональная тренировка по фитнес аэробике, йоге и т.д. (2-3 чел. одновременно) – 1 000 руб.
- Разработка индивидуальной программы тренировок - 500 руб.
- Абонемент «Лучик» - детский фитнес (8 тренировок) - 700 руб.

Приложение 1.3

ДОГОВОР ОБ ОКАЗАНИИ УСЛУГ № ____

г. Москва _____ 20__ г.

ООО «F.I.T.» в лице генерального директора Дерябина Михаила Алексеевича, именуемый в дальнейшем Исполнитель, действующий на основании Свидетельства №305760209600032 выд. Инспекцией Федеральной налоговой службы по Центрального р-на г. Москвы 06 апреля 2005 г., с одной стороны, и _____, именуемое в дальнейшем Заказчик, в лице генерального директора _____, действующего на основании Устава, с другой стороны, а вместе именуемые «Стороны», заключили настоящий Договор о нижеследующем:

1. ПРЕДМЕТ ДОГОВОРА

Заказчик поручает, а Исполнитель принимает на себя оказание услуг, предоставляемых фитнес-клубом «F.I.T.».

2. СТОИМОСТЬ УСЛУГ И ПОРЯДОК РАСЧЕТОВ

2.1. Стоимость оказываемых Исполнителем услуг по настоящему Договору составляет: _____, НДС не облагается и включает в себя _____ на посещение фитнес-клуба «F.I.T.» стоимостью _____ руб.

2.2. Оплата услуг Исполнителя производится Заказчиком в виде предоплаты 50% на расчетный счет не позднее десяти банковских дней после выписки счета, либо в кассу Исполнителя. Оставшиеся 50% уплачиваются Заказчиком Исполнителю в срок до _____ 20__ года.

3. ОБЯЗАННОСТИ СТОРОН

3.1 Исполнитель обязуется оказывать в течение срока действия настоящего договора услуги Заказчику, согласно действующих условий. Предоставление услуг производится согласно режима работы фитнес-клуба «F.I.T.».

3.2. Исполнитель обязуется выполнить работы по оказанию услуг Заказчику.

3.3. Заказчик обязуется соблюдать правила внутреннего распорядка фитнес-клуба «F.I.T.».

4. ОТВЕТСТВЕННОСТЬ СТОРОН. ФОРС-МАЖОР

4.1. За неисполнение или ненадлежащее исполнение обязательств по настоящему Договору Стороны несут ответственность в соответствии с действующим законодательством Российской Федерации.

4.2. Стороны освобождаются от ответственности за полное или частичное неисполнение обязательств по настоящему Договору, если это неисполнение явилось следствием не контролируемых Сторонами обстоятельств, возникших после заключения Договора, которые Стороны не могли ни предвидеть, ни предотвратить разумными мерами (форс-мажор). Возможное неисполнение обязательств по настоящему Договору должно быть непосредственной причинной в связи с указанными выше обстоятельствами.

4.3. При наступлении и прекращении вышеуказанных обстоятельств Сторона, для которой создалась невозможность исполнения своих обязательств по настоящему Договору, должна предпринять все разумные меры, чтобы без промедления известить об этом другую Сторону, предоставив соответствующий документ, выданный компетентным государственным органом.

4.4. Наступление форс-мажорных обстоятельств влечет увеличение срока исполнения Договора на период действия таких обстоятельств (min. 14 дней).

5. ПОРЯДОК РАСТОРЖЕНИЯ ДОГОВОРА

5.1. Договор может быть расторгнут Сторонами досрочно в случаях, предусмотренных действующим законодательством РФ.

6. ПРОЧИЕ УСЛОВИЯ

6.1. Настоящий Договор вступает в силу с момента его подписания сторонами и действует до полного выполнения сторонами своих обязательств.

6.2. Все изменения и дополнения к настоящему Договору являются действительными, если они совершены в письменной форме и подписаны уполномоченными на то представителями сторон.

6.3. Настоящий Договор составлен в двух экземплярах, имеющих одинаковую юридическую силу, по одному экземпляру для каждой из сторон.

6.4. Во всем остальном, что не предусмотрено настоящим Договором, применяются правила, установленные действующим законодательством РФ.

6.5. Споры и разногласия, которые могут возникнуть при исполнении настоящего Договора, должны, по возможности, разрешаться путем переговоров между Сторонами.

6.6 В случае невозможности разрешения споров путем переговоров Стороны решают их в судебном порядке.

7. ЮРИДИЧЕСКИЕ АДРЕСА И РЕКВИЗИТЫ СТОРОН:

ЗАКАЗЧИК:

Ген. директор

м.п. «__» _____ 20__ г.

ИСПОЛНИТЕЛЬ:

ООО «F.I.T.»

111111, г. Москва, пр. Дзержинского, д.34,
кв46 инн: 760211957269
Р/с:40802810577030101356 в Северном банке Сбербанка РФ
К/с:30101811050000000670, БИК:
047888670 Ген. директор
_____ Дерябин М.А.
м.п. «__» _____ 20__ г.

Приложение 1.4

Лист записи на тренировку

«__» _____ 20__ г.

Фитнес зал

17:00 – 18:00 Step + Power Class	...					
Иванов						
Петров						
Сидоров						
...						