

Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Институт повышения квалификации
и переподготовки

Кафедра «Профессиональная переподготовка»

А. Н. Осипенко

УПРАВЛЕНИЕ РАЗРАБОТКОЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

ПРАКТИКУМ
по одноименному курсу
для слушателей специальности переподготовки
1-40 01 73 «Программное обеспечение
информационных систем»
заочной формы обучения

Гомель 2017

УДК 004.4(075.8)
ББК 32.972я73
О-74

*Рекомендовано кафедрой «Профессиональная переподготовка»
Института повышения квалификации
и переподготовки ГГТУ им. П. О. Сухого
(протокол № 3 от 28.11.2016 г.)*

Рецензент: зав. каф. «Информатика» ГГТУ им. П. О. Сухого»
канд. физ.-мат. наук, доц. *Т. В. Тихоненко*

Осипенко, А. Н.
О-74 Управление разработкой программного обеспечения : практикум по одноим. курсу для слушателей специальности переподготовки 1-40 01 73 «Программное обеспечение информационных систем» заоч. формы обучения / А. Н. Осипенко. – Гомель : ГГТУ им. П. О. Сухого, 2017. – 43 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <http://elib.gstu.by>. – Загл. с титул. экрана.

Практикум содержит основные теоретические сведения и задания, необходимые для понимания принципов управления разработкой программного обеспечения, а также для организации эффективной работы менеджера проекта информационной системы.

Издание адресовано слушателям ИПКиП специальности 1-40 01 73 «Программное обеспечение информационных систем».

**УДК 004.4(075.8)
ББК 32.972я73**

© Учреждение образования «Гомельский
государственный технический университет
имени П. О. Сухого», 2017

Оглавление

Предисловие.....	4
1 Лабораторная работа №1. Разработка спецификаций требований к IT-проекту	5
1.1 Разработка диаграммы вариантов использования и редактирование свойств ее элементов	5
1.2 Спецификация требований и рекомендации по написанию эффективных вариантов использования.....	14
1.3 Шаблоны спецификаций требований на ПО.....	23
1.4 Задания для лабораторной работы 1	28
2 Лабораторная работа №2 Управление разработкой ПО в среде Microsoft Project.....	33
2.1 Работа в среде Microsoft Project	33
2.2 Задания для лабораторной работы 2	36
Список использованных источников.....	43

Предисловие

Управление разработкой программного обеспечения является одним из важнейших направлений деятельности в современной информационной сфере. Стремительный рост числа пользователей интернета и открытие большого числа сайтов привело к тому, что создание и поддержка веб-сайтов со все более развитым функционалом стало востребованным направлением ИТ-технологий. Вместе с тем, в данный момент создание программного обеспечения происходит по далеким от совершенства методикам. Особенно актуальными при этом оказываются вопросы постановки задачи автоматизации бизнес-процессов и взаимодействия заказчика и разработчика [1-6]. В связи с этим в предлагаемом практикуме основное внимание уделено разработке спецификаций технического задания проекта с использованием case-технологий (диаграмм вариантов использования [7]), а также качественному управлению разработкой ПО средствами MS Project [8-10]. Настоящее пособие разработано в соответствии с программой дисциплины «Управление разработкой программного обеспечения» для обучения слушателей ИПК и П заочной формы обучения по специальности 1-40 01 73 «Программное обеспечение информационных систем».

Целью практикума является повышение профессионального уровня будущих специалистов по созданию программного обеспечения информационных систем в области менеджмента ИТ-проектов. Задачей практикума является формирование у слушателей современных представлений о разработке ИТ-проектов, привлечении кадров, распределении времени, работе с заказчиками, формированию требований к проекту, а также решению проблем эффективного использования современных инструментов разработки программного обеспечения на основе накопленного к настоящему времени мирового опыта.

1 Лабораторная работа №1. Разработка спецификаций требований к IT-проекту

1.1 Разработка диаграммы вариантов использования и редактирование свойств ее элементов

Работа над моделью в среде IBM Rational Rose [7] начинается с общего анализа проблемы и построения диаграммы вариантов использования, которая отражает функциональное назначение проектируемой программной системы. Для вновь создаваемого проекта можно воспользоваться мастером типовых проектов, если он установлен в данной конфигурации. Мастер типовых проектов доступен из меню **File>New** (Файл>Новый) или при первоначальной загрузке программы IBM Rational Rose 2003. В качестве проекта далее будет рассматриваться модель системы управления банкоматом. Для изменения имени проекта, предложенного программой по умолчанию, следует сохранить модель во внешнем файле на диске, например, под именем ATMmodel.mdl.











Для разработки диаграммы вариантов использования модели в среде IBM Rational Rose 2003 необходимо активизировать соответствующую диаграмму в окне диаграммы. Это можно сделать следующими способами:

- раскрыть представление вариантов использования **Use Case View** в браузере проекта и дважды щелкнуть на пиктограмме **Main** (Главная);
- с помощью операции главного меню **Browse>Use Case Diagram** (Браузер>Диаграмма вариантов использования).

При этом появляется новое окно с чистым рабочим листом диаграммы вариантов использования и специальная панель инструментов, содержащая кнопки с изображением графических элементов, необходимых для разработки диаграммы вариантов использования. Назначение отдельных кнопок данной панели можно узнать также из всплывающих подсказок, которые появляются, если подвести и задержать на некоторое время указатель мыши над той или иной кнопкой (таблица 1.1). На специальной панели инструментов по умолчанию присутствует только часть кнопок с пиктограммами элементов, которые могут быть использованы для построения диаграммы. Добавить кнопки с пиктограммами других графических элементов, например, таких как бизнес-вариант использования (business use case), бизнес-актер (business actor), сотрудник (business worker), или удалить ненужные кнопки можно с помощью настройки

специальной панели инструментов.

Таблица 1.1 – Назначение кнопок специальной панели инструментов для диаграммы вариантов использования

Графическое изображение	Всплывающая подсказка	Назначение кнопки
	Selection Tool	Превращает изображение курсора в форму стрелки для последующего выделения элементов на диаграмме
	Text Box	Добавляет на диаграмму текстовую область
	Note	Добавляет на диаграмму примечание
	Anchor Note to Item	Добавляет на диаграмму связь примечания с соответствующим графическим элементом диаграммы
	Package	Добавляет на диаграмму пакет
	Use Case	Добавляет на диаграмму вариант использования
	Actor	Добавляет на диаграмму актера
	Unidirectional Association	Добавляет на диаграмму направленную ассоциацию
	Dependency or Instantiates	Добавляет на диаграмму отношение зависимости
	Generalization	Добавляет на диаграмму отношение обобщения

Открыть диалоговое окно настройки специальных панелей инструментов можно с помощью операции главного меню: **Tools>Options** (Инструменты>Параметры), раскрыв вкладку **Toolbars** (Панели инструментов) и нажав соответствующую кнопку (например, **Use Case diagram**) в группе опций **Customize Toolbars** (Настройка панелей инструментов). Это окно настройки также можно открыть с помощью операции контекстного меню **Customize** (Настройка) при позиционировании курсора на специальной панели инструментов (рисунок 1.1).

Для добавления необходимых кнопок на панель следует выделить их в левом окне со списком пиктограмм графических элементов, после чего нажать кнопку **Добавить** в центре диалогового окна. Для удаления ненужных кнопок с панели инструментов следует выделить их в правом окне со списком пиктограмм графических элементов, после чего нажать кнопку **Удалить** в центре диалогового окна. Для

восстановления набора пиктограмм по умолчанию можно нажать кнопку **Сброс**. После настройки специальной панели инструментов соответствующее окно следует закрыть нажатием на кнопку **Закреть**.

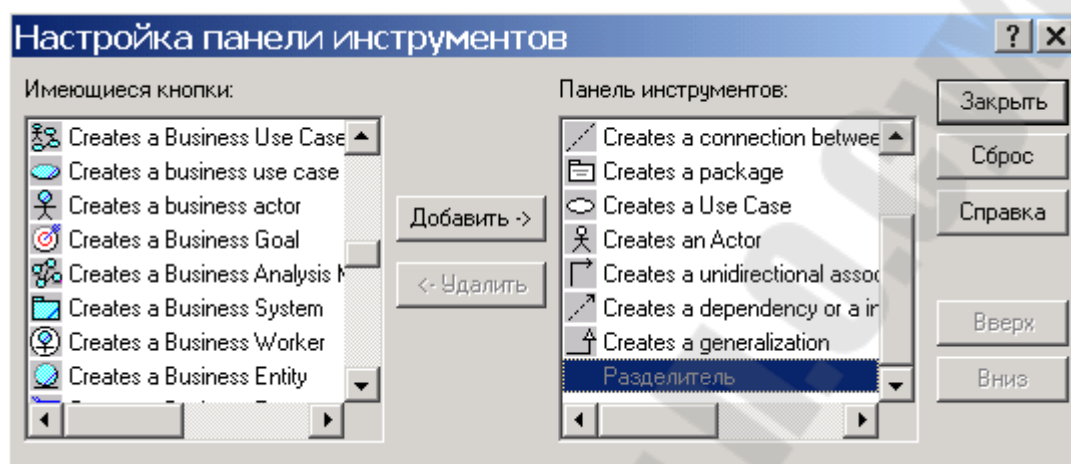


Рисунок 1.1 – Диалоговое окно настройки специальной панели инструментов для диаграммы вариантов использования

Добавление актера на диаграмму вариантов использования и редактирование его свойств.

Для добавления актера на диаграмму варианта использования нужно с помощью левой кнопки мыши нажать кнопку с изображением пиктограммы актера на специальной панели инструментов, отпустить левую кнопку мыши и щелкнуть левой кнопкой мыши на свободном месте рабочего листа диаграммы. На диаграмме появится изображение актера с маркерами изменения его геометрических размеров и предложенным программой именем по умолчанию NewClass. Для разрабатываемой модели банкомата предложенное программой имя актера следует изменить на Клиент Банкомата (рисунок 1.2).

Чтобы изменить расположение изображения графического элемента модели, следует щелчком левой кнопки мыши выделить его в рабочей области диаграммы, и, не отпуская левой кнопки, переместить в нужное место диаграммы. При этом выделенный элемент визуально отличается от остальных наличием маркеров изменения его геометрических размеров в форме небольших черных квадратов. Более точное перемещение элемента можно осуществить с помощью стрелок: ←, ↑, →, ↓ на клавиатуре.

Имя размещенного на диаграмму элемента разработчик может изменить либо сразу после добавления элемента на диаграмму, либо в ходе последующей работы над проектом. Для любого графического элемента модели по щелчку правой кнопкой мыши на выбранном элементе вызывается контекстное меню данного элемента, среди

операций которого имеется пункт **Open Specification** (Открыть спецификацию). В этом случае появляется дополнительное диалоговое окно со специальными вкладками, в поля ввода которых можно занести всю информацию по данному элементу. Для добавленного актера Клиент Банкомата окно спецификации свойств выглядит следующим образом (рисунок 1.3).

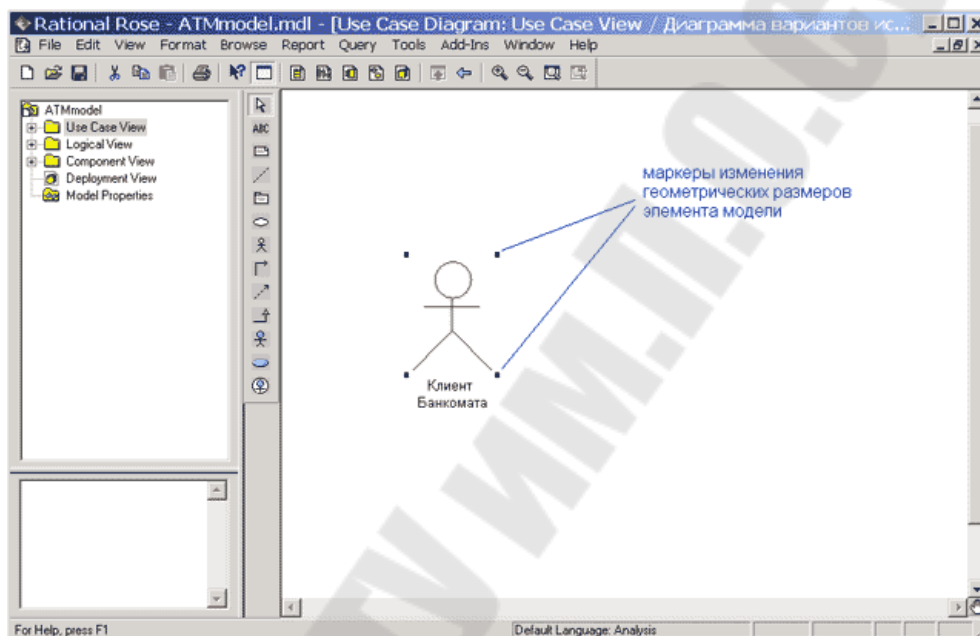


Рисунок 1.2 – Диаграмма вариантов использования после добавления на нее актера

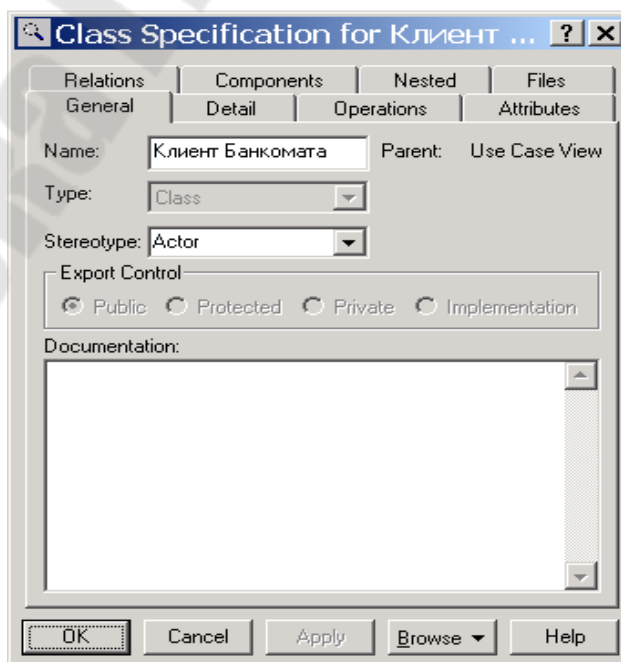


Рисунок 1.3 – Диалоговое окно спецификации свойств актера Клиент Банкомата

Следует отметить, что открыть диалоговое окно спецификации свойств любого элемента модели можно также двойным щелчком левой кнопкой мыши на графическом изображении этого элемента на диаграмме. Хотя в среде IBM Rational Rose актер является классом, для него некорректно специфицировать атрибуты и операции, поскольку актер является внешней по отношению к разрабатываемой системе сущностью.

Для актера Клиент Банкомата можно уточнить его назначение в модели. С этой целью следует изменить его стереотип и добавить текст документации. Для изменения стереотипа во вложенном списке **Stereotype** нужно выбрать строку **Business Actor** (бизнес-актер). Для добавления текста документации в секцию **Documentation** следует ввести текст: "Любое физическое лицо, пользующееся услугами банкомата" и нажать кнопку **Apply** (Применить) или **OK**.

Добавление и редактирование варианта использования

Для добавления варианта использования на диаграмму нужно с помощью левой кнопки мыши нажать кнопку с изображением варианта использования на специальной панели инструментов, отпустить левую кнопку мыши и щелкнуть левой кнопкой мыши на свободном месте диаграммы. На диаграмме появится изображение варианта использования с маркерами изменения его геометрических размеров и предложенным программой именем по умолчанию NewUseCase. Для разрабатываемой модели банкомата предложенное программой имя варианта использования следует изменить на Снятие наличных по кредитной карточке (рисунок 1.4).

Для уточнения свойств данного варианта использования следует открыть диалоговое окно спецификации его свойств, например, с помощью двойного щелчка левой кнопкой мыши на изображении этого элемента на диаграмме. Для изменения стереотипа во вложенном списке **Stereotype** нужно выбрать строку **Business Use Case**.

Добавление ассоциации

Для добавления ассоциации между актером и вариантом использования на диаграмму нужно с помощью левой кнопки мыши нажать на специальной панели инструментов кнопку с изображением пиктограммы направленной ассоциации, отпустить левую кнопку мыши, щелкнуть левой кнопкой мыши на изображении актера на диаграмме и отпустить ее на изображении варианта использования. В результате этих действий на диаграмме появится изображение ассоциации, соединяющей актера с вариантом использования (рисунок 1.5). При необходимости можно сделать направленную ассоциацию ненаправленной, для чего следует воспользоваться диалоговым окном

свойств ассоциации. Открыть это окно можно, например, двойным щелчком на изображении линии ассоциации на диаграмме, после чего убрать отметку строки выбора **Navigable** (Навигация) на вкладке **Role A Detail** (Детальные свойства конечной точки ассоциации A).

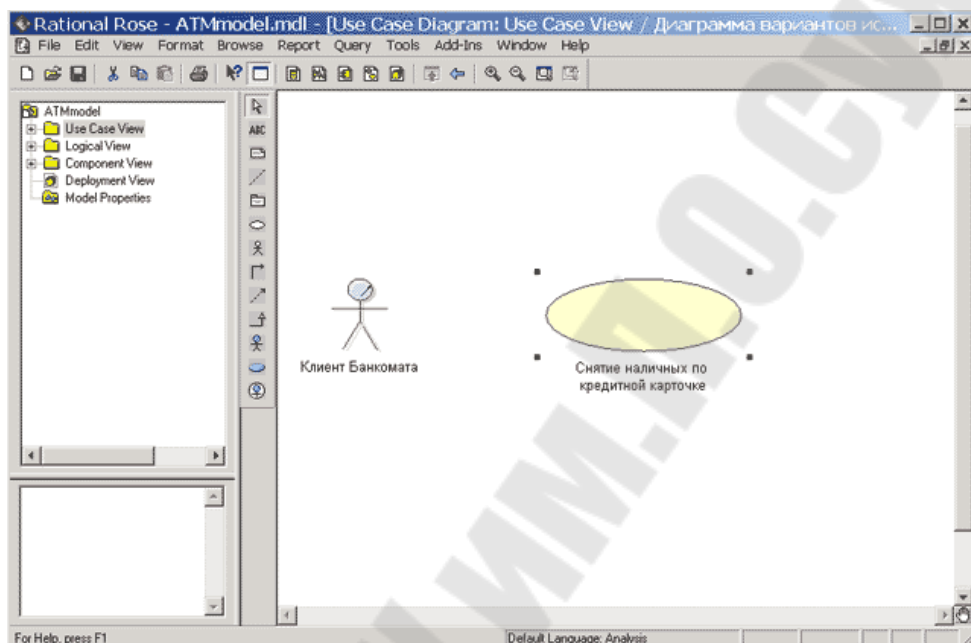


Рисунок 1.4 – Диаграмма вариантов использования после добавления на нее варианта использования

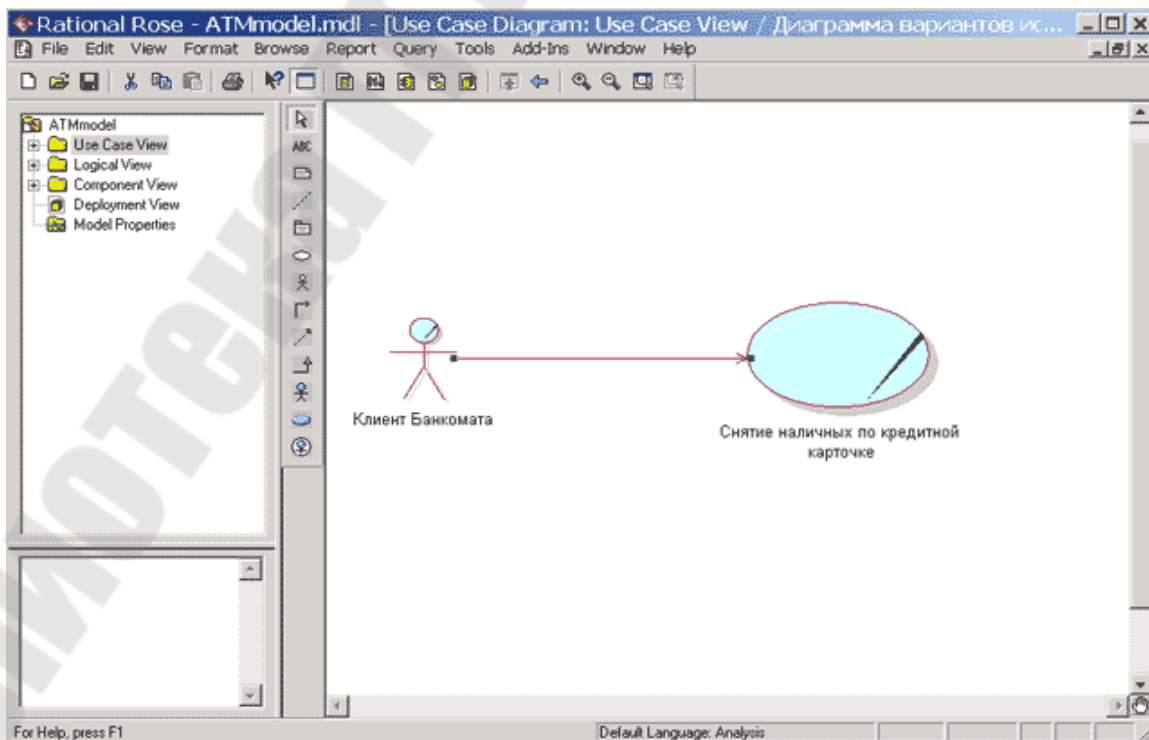


Рисунок 1.5 – Диаграмма вариантов использования после добавления на нее направленной ассоциации

Добавление отношения зависимости и редактирование его свойств

Для добавления отношения зависимости между двумя вариантами использования на диаграмму необходимо предварительно рассмотренным выше способом добавить второй вариант использования с именем Проверка ПИН-кода. После этого с помощью левой кнопки мыши нажать кнопку с изображением пиктограммы зависимости на специальной панели инструментов, отпустить левую кнопку мыши, щелкнуть левой кнопкой мыши на изображении варианта использования Снятие наличных по кредитной карточке и отпустить ее на изображении варианта использования Проверка ПИН-кода. В результате этих действий на диаграмме появится изображение отношения зависимости, которое соединяет два выбранных варианта использования.

Поскольку вариант использования Проверка ПИН-кода выполняется всегда, для добавленного отношения зависимости дополнительно следует указать текстовый стереотип `<<include>>`. Выполнить это можно уже известным способом с помощью диалогового окна спецификации свойств этого отношения и выбора нужного стереотипа из предлагаемого списка.

После задания для данного отношения зависимости стереотипа `<<include>>` текст этого стереотипа в угловых скобках появится рядом с изображением пунктирной линии зависимости, связывающей соответствующие варианты использования (рисунок 1.6). С целью лучшей визуализации диаграммы текстовую область стереотипа можно переместить в нужное место диаграммы.

Аналогичным образом могут быть добавлены на диаграмму вариантов использования отношения зависимости со стереотипом `<<extend>>`, которые применяются для моделирования исключений при выполнении отдельных вариантов использования.

Окончательное построение диаграммы вариантов использования

К отдельному варианту использования можно добавить текстовый файл с описанием сценария его выполнения. Для этого необходимо выделить этот вариант использования в браузере проекта и выполнить операцию контекстное меню: **New>File** (Новый>Файл). В результате этого будет вызвано стандартное окно открытия файла, в котором необходимо задать имя предварительно созданного с помощью офисной программы MS Word добавляемого файла. После нажатия кнопки **Открыть** пиктограмма добавленного файла появится в браузере проекта ниже соответствующего варианта использования. В

последующем можно вернуться к редактированию этого файла сценария, выполнив двойной щелчок на этой пиктограмме. При этом файл сценария будет открыт в соответствующем приложении - в текстовом процессоре MS Word.

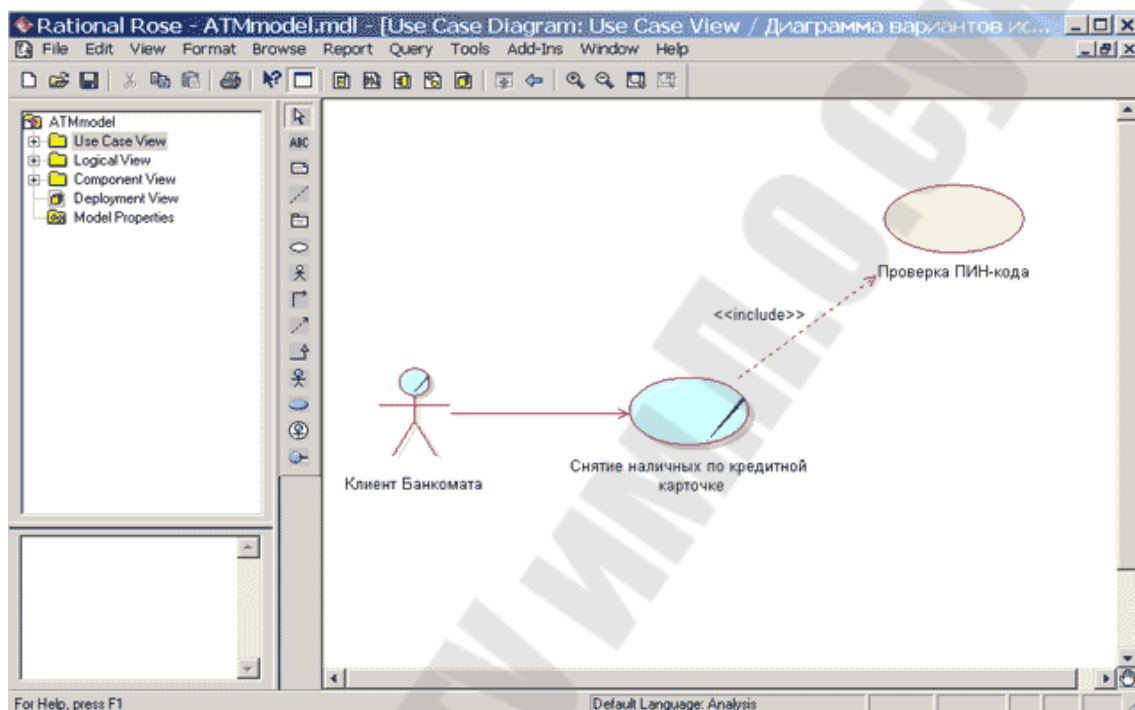


Рисунок 1.6 – Диаграмма вариантов использования после добавления на нее отношения зависимости

Для окончательного построения диаграммы варианта использования для рассматриваемой модели банкомата следует выполнить следующие действия:

1. Добавить актера с именем Банк, для которого выбрать стереотип **Service** (Сервис), означающий, что банкомат использует некоторые услуги Банка в качестве сервиса.
2. Добавить вариант использования Получение справки о состоянии счета, для которого выбрать стереотип **Business Use Case** (Бизнес-вариант использования).
3. Добавить вариант использования Блокирование кредитной карточки.
4. Добавить направленную ассоциацию от бизнес-актера Клиент Банкомата к варианту использования Получение справки о состоянии счета.
5. Добавить направленную ассоциацию от варианта использования Снятие наличных по кредитной карточке к сервису Банк.
6. Добавить направленную ассоциацию от варианта использования Получение справки о состоянии счета к сервису Банк.

7. Добавить отношение зависимости со стереотипом <<include>>, направленное от варианта использования Получение справки о состоянии счета к варианту использования Проверка ПИН-кода.
8. Добавить отношение зависимости со стереотипом <<extend>>, направленное от варианта использования Блокирование кредитной карточки к варианту использования Проверка ПИН-кода.

Выполнить эти действия предлагается самостоятельно. При этом отношение зависимости со стереотипом <<extend>> на данной диаграмме означает следующее. Вариант использования Блокирование кредитной карточки будет выполняться только в том случае, если в результате проверки ПИН-кода будет установлено, что соответствующая кредитная карточка утрачена ее владельцем или признана недействительной. Построенная таким образом диаграмма вариантов использования будет иметь следующий вид (рисунок 1.7).

Напомним, что диаграмма вариантов использования является высокоуровневым концептуальным представлением модели, поэтому она не должна содержать слишком много вариантов использования и актеров. В последующем построенная диаграмма может быть изменена посредством добавления новых элементов, таких как варианты использования и актеры, или их удаления.

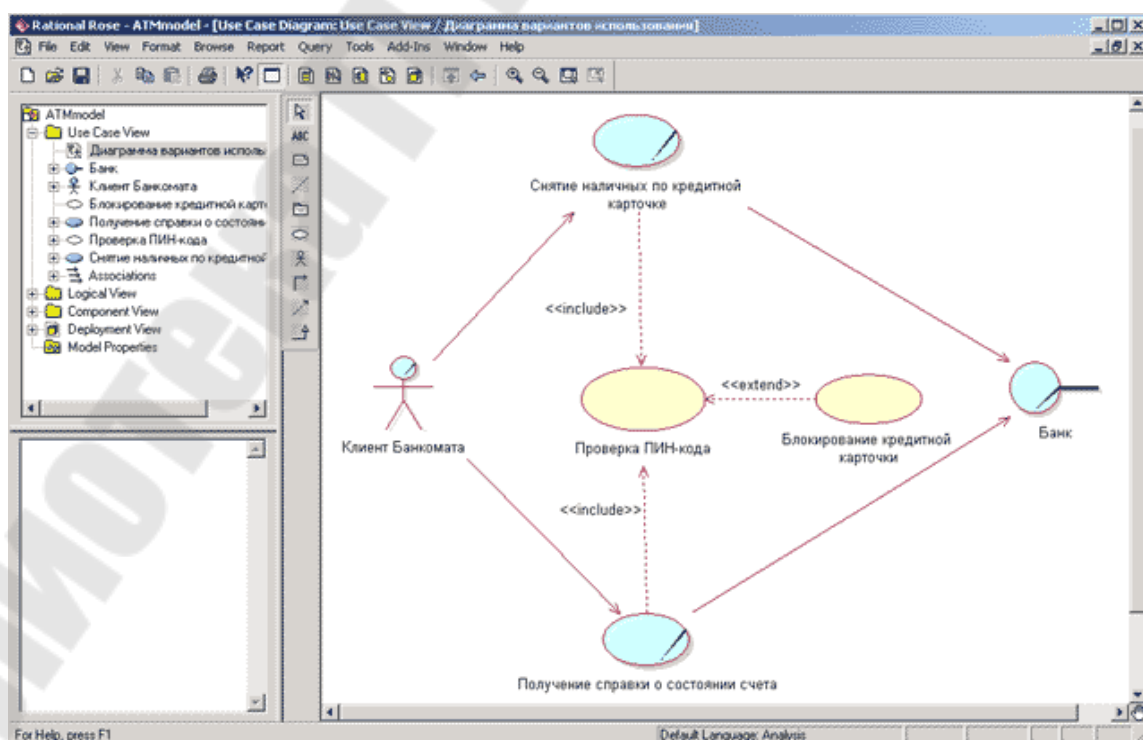


Рисунок 1.7 – Окончательный вид диаграммы вариантов использования для разрабатываемой модели банкоматов

Для удаления любого графического элемента с диаграммы его следует выделить на диаграмме и нажать клавишу **Delete** на клавиатуре. При этом выделенный элемент будет удален с активной диаграммы, но не из модели. Для удаления элемента не только из диаграммы, но и из модели проекта необходимо выделить удаляемый элемент на диаграмме и воспользоваться операцией главного меню **Edit>Delete from Model** (Редактирование>Удалить из модели). Для этой же цели служит комбинация клавиш быстрого доступа: **Ctrl+D**.

После окончания сеанса работы над проектом выполненную работу необходимо сохранить в файле проекта с расширением ".MDL". Это можно сделать через меню **File>Save** (Файл>Сохранить) или **File>Save As** (Файл>Сохранить как). При этом вся информация о проекте, включая диаграммы и спецификации элементов, будет сохранена в одном файле.

1.2 Спецификация требований и рекомендации по написанию эффективных вариантов использования

Формализация функциональных требований к системе с помощью диаграммы вариантов использования

Отдельные варианты использования могут применяться как для спецификации требований к проектируемой системе, так и для документирования процесса поведения имеющейся системы. Кроме этого, варианты использования неявно специфицируют требования, определяющие особенности взаимодействия пользователей с системой, которые специфицируют возможность корректной работы с предоставляемыми данной системой сервисами.

Требование (requirement) – желательное свойство, характеристика или условие, которым должна удовлетворять система в процессе своей эксплуатации.

Применительно к программным системам предложена следующая классификация требований, которая получила название модели FURPS+, что соответствует первым буквам соответствующих категорий требований на английском языке:

- функциональные требования (Functionality);
- требования удобства использования (Usability);
- требования надежности (Reliability);
- требования производительности (Performance);
- требования возможности сопровождения (Supportability).

При этом символом "+" обозначены дополнительные условия, к

которым относятся:

- проектные ограничения;
- требования управления системой;
- требования к графическому интерфейсу пользователя;
- физические требования;
- юридические требования.

Центральное место среди указанных требований занимают функциональные, которые специфицируют особенности реализации отдельных бизнес-процессов моделируемой системы. В контексте моделей языка UML именно функциональные требования должны служить исходной информацией для построения диаграмм вариантов использования. Однако графических средств языка UML на практике оказывается недостаточно для спецификации функциональных требований.

Следует отметить, что одним из требований языка UML является самодостаточность диаграмм для представления информации о моделях проектируемых систем. Однако большинство разработчиков и экспертов согласны с тем, что изобразительных средств языка UML явно не хватает для того, чтобы учесть на диаграммах вариантов использования особенности функционального поведения сложной системы. С этой целью рекомендуется дополнять этот тип диаграмм текстовыми сценариями, которые уточняют или детализируют последовательность действий, совершаемых системой при выполнении ее вариантов использования.

Сценарий (scenario) – определенная последовательность действий, которая описывает действия актеров и поведение моделируемой системы в форме обычного текста.

В контексте языка UML сценарий используется для дополнительной иллюстрации взаимодействия актеров и вариантов использования. Предлагаются различные способы представления или написания подобных сценариев. Один из таких шаблонов рассматривается ниже и может быть рекомендован слушателям для применения на начальных этапах концептуального моделирования (таблица 1.2).

При написании сценариев вариантов использования важно понимать, что текст сценария должен дополнять или уточнять диаграмму вариантов использования, но не заменять ее полностью. В противном случае будут потеряны достоинства визуального представления моделей.

Таблица 1.2 – Шаблон для написания сценария отдельного варианта использования

Главный раздел	Раздел "Типичный ход событий"	Раздел "Исключения"	Раздел "Примечания"
Имя варианта использования	Типичный ход событий, приводящий к успешному выполнению варианта использования	Исключение № 1	Примечания № 1
Актеры		Исключение № 2	Примечания № 2
Цель	
Краткое описание			
Тип			
Ссылки на другие варианты использования			Исключение № N

Построение диаграммы вариантов использования – самый первый этап процесса объектно-ориентированного анализа и проектирования, цель которого – представить совокупность функциональных требований к поведению проектируемой системы. **Спецификация требований** к проектируемой системе в форме диаграммы вариантов использования и дополнительных сценариев может представлять собой самостоятельную модель, которая в языке UML получила название модели вариантов использования и имеет свое специальное стандартное имя или стереотип <<useCaseModel>>.

Все заданные в этой модели требования допустимо представить в виде общей модели системы, которая может быть оформлена как отдельный пакет Система. Этот пакет в свою очередь может представлять собой иерархию пакетов, на самом верхнем уровне которых содержится множество классов, реализующих базовые варианты использования проектируемой системы. При этом пакет системы самого верхнего уровня может быть дополнительно помечен стереотипом <<topLevelPackage>>.

Особенности спецификации функциональных требований на диаграмме вариантов использования

Для иллюстрации особенностей спецификации функциональных требований на диаграмме вариантов использования снова вернемся к модели обычного банкомата. Особенность отечественных банкоматов состоит в том, что в них отсутствует возможность перевода средств с одного счета на другой. Рассматриваемая система имеет двух актеров, один из которых является клиентом банкомата, а другой – Банком, который осуществляет выполнение соответствующих транзакций.

Каждый из этих актеров взаимодействует с банкоматом, хотя главный актер Клиент, поскольку именно он инициирует функциональность банкомата.

Основные функциональные требования к банкомату заключаются в предоставлении клиенту возможности снятия наличных по кредитной карточке и получении справки о состоянии счета. Именно эти функциональные требования специфицируются отдельными вариантами использования, которые служат ключевыми элементами соответствующей концептуальной модели. Поскольку для выполнения этих вариантов использования необходимо аутентифицировать кредитную карточку, они оба обращаются к дополнительному сервису "Проверка ПИН-кода кредитной карточки". Как следует из существа выдвигаемых к банкомату функциональных требований, этот сервис может выступать в качестве отдельного варианта использования разрабатываемой диаграммы и соединяться с первыми двумя вариантами отношением включения. Соответствующая диаграмма вариантов использования может включать в себя только указанных двух актеров и три варианта использования (рисунок 1.8).

На следующем этапе разработки модели вариантов использования для рассматриваемой системы банкомата следует дополнить данную диаграмму текстовым сценарием, написанным на основе предложенного ранее шаблона (таблица 1.2). Этот сценарий будет дополнять диаграмму, раскрывая содержание и логическую последовательность отдельных действий, которые выполняются системой и актерами в процессе снятия наличных по кредитной карточке. В этом случае сценарий удобно представить в виде трех таблиц, каждая из которых описывает отдельный раздел шаблона.

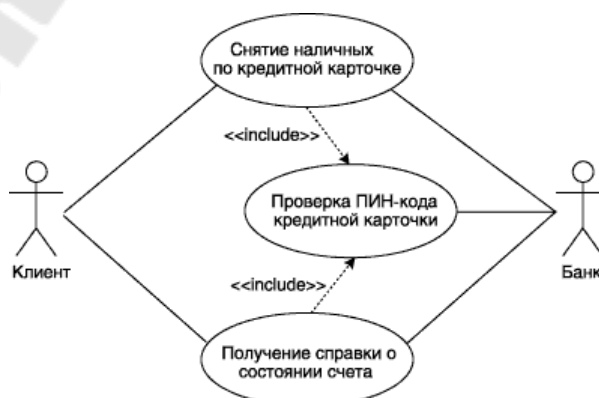


Рисунок 1.8 – Диаграмма вариантов использования для модели банкомата

В главном разделе сценария (таблица 1.3) указывается имя

рассматриваемого варианта использования, имена взаимосвязанных с ним актеров, цель выполнения варианта, условный тип и ссылки на другие варианты использования.

Таблица 1.3. Главный раздел сценария выполнения варианта использования "Снятие наличных по кредитной карточке"

Вариант использования	Снятие наличных по кредитной карточке
Актеры	Клиент, Банк
Цель	Получение требуемой суммы наличными
Краткое описание	Клиент запрашивает требуемую сумму. Банкомат обеспечивает доступ к счету клиента. Банкомат выдает клиенту наличные.
Тип	Базовый
Ссылки на другие варианты использования	Включает в себя ВИ: <ul style="list-style-type: none"> • Проверка ПИН-кода кредитной карточки • Идентифицировать кредитную карточку

В следующем разделе сценария (таблица 1.4) описывается последовательность действий, приводящая к успешному выполнению рассматриваемого варианта использования. При этом инициатором действий должен выступать актер Клиент. Для удобства последующих ссылок каждое действие помечается порядковым номером в последовательности.

В третьем разделе сценария (таблица 1.5) описывается последовательность действий, выполняемых при возникновении исключительных ситуаций или исключений. Можно дополнить данный сценарий, аналогичным образом описав не только варианты использования "Получение справки о состоянии счета" и "Проверка Пин-кода кредитной карточки", но и рассмотрев другие исключения, например отказ клиента от получения наличных после проверки ПИН-кода и т.п. При этом полнота сценариев и модели вариантов использования будут определяться теми функциональными требованиями, которые сформулированы в рамках конкретного проекта разработки соответствующего банкомата.

Отдельные небольшие по своему объему сценарии могут быть размещены на диаграмме в форме примечаний.

Примечание (note) предназначено для включения в модель произвольной текстовой информации, имеющей непосредственное отношение к контексту разрабатываемого проекта.

Таблица 1.4. Раздел Типичный ход событий сценария выполнения варианта использования "Снятие наличных по кредитной карточке"

Действия актеров	Отклик системы
1. Клиент вставляет кредитную карточку в устройство чтения банкомата Исключение №1: Кредитная карточка недействительна	2. Банкомат проверяет кредитную карточку 3. Банкомат предлагает ввести ПИН-код
4. Клиент вводит персональный PIN-код Исключение №2: Клиент вводит неверный ПИН-код	5. Банкомат проверяет ПИН-код 6. Банкомат отображает опции меню
7. Клиент выбирает снятие наличных со своего счета	8. Система делает запрос в Банк и выясняет текущее состояние счета клиента 9. Банкомат предлагает ввести требуемую сумму
10. Клиент вводит требуемую сумму 11. Банк проверяет введенную сумму Исключение №3: Требуемая сумма превышает сумму на счете клиента	12. Банкомат изменяет состояние счета клиента, выдает наличные и чек
13. Клиент получает наличные и чек	14. Банкомат предлагает клиенту забрать кредитную карточку
15. Клиент получает свою кредитную карточку	16. Банкомат отображает сообщение о готовности к работе

Таблица 1.5. Раздел Исключения сценария выполнения варианта использования "Снятие наличных по кредитной карточке"

Исключение №1. Кредитная карточка недействительна или неверно вставлена	
Действия актера	Отклик системы
	3. Банкомат отображает информацию о неверно вставленной кредитной карточке 14. Банкомат возвращает клиенту его кредитную карточку
15. Клиент получает свою кредитную карточку	
Исключение №2. Клиент вводит неверный ПИН-код	
	6. Банкомат отображает информацию о неверном ПИН-коде
4. Клиент вводит новый ПИН-код	
Исключение №3. Требуемая сумма превышает сумму на счете клиента	
	12. Банкомат отображает информацию о превышении кредита
10. Клиент вводит новую требуемую сумму	

В качестве такой информации могут быть комментарии разработчика (например, дата и версия разработки диаграммы или ее отдельных компонентов), ограничения (например, на значения отдельных связей или экземпляры сущностей) и помеченные значения. Применительно к диаграммам вариантов использования примечание может иметь уточняющую информацию, относящуюся к контексту тех или иных вариантов использования.

Графически примечания на всех типах диаграмм обозначаются прямоугольником с "загнутым" верхним правым уголком (рисунок 1.9). Собственно текст примечания размещается внутри этого прямоугольника. Примечание может относиться к любому элементу диаграммы, в этом случае их соединяет пунктирная линия. Если примечание относится к нескольким элементам, то от него проводятся соответственно, несколько линий. Как уже отмечалось, примечания могут присутствовать не только на диаграмме вариантов использования, но и на других канонических диаграммах.

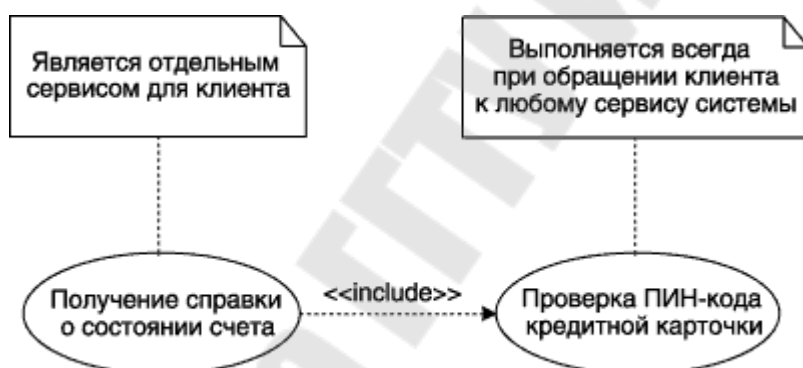


Рисунок 1.9 – Примеры примечаний на диаграммах вариантов использования

Рекомендации по разработке диаграмм вариантов использования

Как было отмечено ранее, одно из главных назначений диаграммы вариантов использования заключается в формализации функциональных требований к системе. Диаграмма вариантов использования может служить основой для согласования с заказчиком функциональных требований к системе на ранней стадии проектирования. Любой из базовых вариантов использования в последующем может быть подвергнут декомпозиции на частные варианты использования. При этом рекомендуется, чтобы общее количество актеров в модели не превышало 20, а вариантов использования – 50. В противном случае модель теряет свою

наглядность и, возможно, заменяет собой одну из некоторых других диаграмм.

Для разработки диаграммы вариантов использования рекомендуется некоторая последовательность действий:

- определить главных или первичных и второстепенных актеров;
- определить цели главных актеров по отношению к системе;
- сформулировать основные варианты использования, которые специфицируют функциональные требования к системе;
- упорядочить варианты использования по степени убывания риска их реализации;
- рассмотреть все базовые варианты использования в порядке убывания их степени риска;
- выделить участников, интересы, предусловия и постусловия выполнения выбранного варианта использования;
- написать успешный сценарий реализации выбранного варианта использования;
- определить исключения или неуспех в выполнении сценария варианта использования;
- написать сценарии для всех исключений;
- выделить общие варианты использования и изобразить их взаимосвязи с базовыми со стереотипом <<include>>;
- выделить варианты использования для исключений и изобразить их взаимосвязи с базовыми со стереотипом <<extend>>;
- проверить диаграмму на отсутствие дублирования вариантов использования и актеров.

Семантика построения диаграммы вариантов использования должна определяться следующими особенностями рассмотренных выше элементов модели. Отдельный экземпляр варианта использования по своему содержанию является выполнением последовательности действий, которая инициализируется посредством экземпляра сообщения от экземпляра актера. В качестве отклика или ответной реакции на сообщение актера выполняется последовательность действий, установленная для данного варианта использования. При этом актеры могут генерировать новые сообщения для инициирования вариантов использования.

Подобное взаимодействие будет продолжаться до тех пор, пока не закончится выполнение требуемой последовательности действий экземпляром варианта использования, и указанный в модели экземпляр актера не получит требуемый экземпляр сервиса. Окончание взаимодействия означает отсутствие инициализации сообщений от актеров для базовых вариантов использования.

Варианты использования могут быть дополнительно специфицированы примечаниями с текстом, которые в последующем могут стать прототипами операций и методов совместно с атрибутами. Дальнейшая разработка моделей связана с реализацией вариантов использования в виде графа деятельности, посредством конечного автомата или любого другого механизма логического представления поведения, включающего предусловия и постусловия. Взаимодействие между вариантами использования и актерами может уточняться на диаграмме кооперации, когда описываются взаимосвязи между системой, содержащей эти варианты использования, и окружением или внешней средой этой системы.

В случае, когда для представления иерархической структуры проектируемой системы применяются подсистемы, система может быть определена в виде вариантов использования на всех уровнях. Отдельные подсистемы или классы могут реализовывать собственные варианты использования. При этом наиболее общий или абстрактный вариант использования в последующем может уточняться множеством частных вариантов использования, каждый из которых будет определять сервис элемента модели, содержащийся в сервисе исходной системы. В этом контексте общий вариант использования может рассматриваться как суперсервис для уточняющих его вариантов использования, которые, в свою очередь, могут рассматриваться как подсервисы исходного варианта использования.

Отдельные варианты использования нижнего уровня могут участвовать в нескольких кооперациях, т. е. играть определенную роль при выполнении сервисов нескольких вариантов верхнего уровня. Для отдельных таких коопераций могут быть определены соответствующие роли актеров, взаимодействующих с конкретными вариантами использования нижнего уровня. Эти роли будут играть актеры нижнего уровня модели системы. Некоторые из таких актеров могут быть актерами верхнего уровня, но это не противоречит принятым в языке UML семантическим правилам построения диаграмм вариантов использования.

Однако следует заметить, что структура элемента-контейнера не может быть представлена вариантами использования, поскольку они могут представлять только функциональность отдельных элементов модели. Подчиненные варианты использования могут только кооперироваться для совместного выполнения суперсервиса варианта использования верхнего уровня. Эта кооперация также представляется на диаграмме кооперации в виде совместных действий отдельных элементов модели.

Окружение вариантов использования нижнего уровня – самостоятельный элемент модели, который, в свою очередь, может содержать другие элементы модели, определенные для этих вариантов использования. Таким образом, с точки зрения общего представления концептуальной модели взаимодействие между вариантами использования нижнего уровня определяет результат выполнения сервиса варианта верхнего уровня.

Реализация варианта использования зависит от типа элемента модели, в котором он определен. Например, варианты использования моделируемой программной системы могут быть реализованы посредством операций классов модели. Применительно к бизнес-системам варианты использования могут реализоваться сотрудниками этой системы. Во всех случаях элементы системы должны взаимодействовать друг с другом для совместного обеспечения требуемого поведения и выполнения вариантов использования модели.

Если в качестве моделируемой сущности выступает подсистема нижнего уровня, то отдельные пользователи вариантов использования этой подсистемы моделируются актерами. Такие актеры, в свою очередь, могут являться внутренними сотрудниками по отношению к моделируемым системам верхних уровней, что зачастую в явном виде не указываются на диаграммах. Эти элементы модели могут содержаться в других пакетах или подсистемах. В последнем случае роли актеров определяются в том пакете, к которому относится соответствующая подсистема.

Если варианты использования применяются для спецификации части системы, то они будут эквивалентны соответствующим вариантам использования в модели подсистемы соответствующего пакета. При этом важно понимать, что все функциональные требования к системе должны быть явно специфицированы на диаграмме вариантов использования, и никаких других сервисов, которые отсутствуют на данной диаграмме, проектируемая система не может выполнять по определению. Более того, если для моделирования поведения системы используются сразу несколько моделей, то множество вариантов использования всех пакетов системы должно быть эквивалентно множеству вариантов использования модели в целом.

1.3 Шаблоны спецификаций требований на ПО

Существуют различные государственные, отраслевые и

корпоративные стандарты. Наиболее распространенные в странах СНГ:

- IEEE 830-1998 «Recommended Practice for Software Requirements Specifications» (описывает, как спецификацию формировать и представлять с помощью разных шаблонов);
- ГОСТ 34.602–89 «Техническое задание на создание автоматизированной системы».

Шаблон не должен являться догмой (если это не требование заказчика). Следует при необходимости модифицировать шаблоны в соответствии с природой и потребностями проекта.

Наиболее полезным в настоящее время является документ IEEE Guide for Developing – System Requirements Specifications (SRS), описывающий, как необходимо создавать спецификацию требований.

Шаблон спецификации требований (IEEE 830-1998). Общая часть (разделы 1 и 2).

1 Введение

1.1 Назначение

1.2 Область действия

1.3 Определения, акронимы и сокращения

1.4 Публикации

1.5 Краткий обзор

2 Общее описание

2.1 Перспектива изделия

2.2 Функция изделия

2.3 Характеристики пользователей

2.4 Ограничения

2.5 Допущения и зависимости

2.6 Разделение требований

**Шаблон спецификации требований (IEEE 830-1998).
Специфические требования (раздел 3).**

3. Специфические требования

3.1. Внешние интерфейсы

3.2. Функции системы

3.3. Требования к рабочим характеристикам

3.4. Логические требования к базе данных

3.5. Проектные ограничения

3.6. Атрибуты системы программного обеспечения
(нефункциональные требования)

Приведем семь наиболее распространенных шаблонов раздела 3 «Специфические требования», приведенных в стандарте SRS.

Шаблон раздела 3 SRS (по режимам. V1)

- 3.1 Требования к внешним интерфейсам
 - 3.1.1 Интерфейсы пользователя
 - 3.1.2 Аппаратные интерфейсы
 - 3.1.3 Интерфейсы программного обеспечения
 - 3.1.4 Интерфейсы связи
- 3.2 Функциональные требования
 - 3.2.1 Режим 1
 - 3.2.1.1 Функциональные требования 1.1
 - ...
 - 3.2.1.n Функциональные требования 1. n
 - 3.2.2 Режим 2
 - ...
 - 3.2.m Режим m
- 3.3 Требования к рабочим характеристикам
- 3.4 Проектные ограничения
- 3.5 Атрибуты системы программного обеспечения
- 3.6 Другие требования

Шаблон раздела 3 SRS (по режимам. V2)

- 3.1 Функциональные требования
 - 3.1.1 Режим 1
 - 3.1.1.1 Внешние интерфейсы
 - 3.1.1.1.1 Интерфейсы пользователя
 - 3.1.1.1.2 Аппаратные интерфейсы
 - 3.1.1.1.3 Интерфейсы программного обеспечения
 - 3.1.1.1.4 Интерфейсы связи
 - 3.1.1.2 Функциональные требования
 - 3.1.1.2.1 Функциональные требования 1
 - ...
 - 3.1.1.2.n Функциональные требования n
 - 3.1.2 Режим 2
 - ...
 - 3.1.m Режим m
- 3.2 Проектные ограничения
- 3.3 Атрибуты системы программного обеспечения
- 3.4 Другие требования

Шаблон раздела 3 SRS (по классам пользователей)

- 3.1 Внешние интерфейсы
 - 3.1.1 Интерфейсы пользователя
 - 3.1.2 Аппаратные интерфейсы
 - 3.1.3 Интерфейсы программного обеспечения
 - 3.1.4 Интерфейсы связи

- 3.2 Функциональные требования
 - 3.2.1 Класс пользователей 1
 - 3.2.1.1 Функциональные требования 1.1
 - ...
 - 3.2.1.n Функциональные требования 1. n
 - 3.2.2 Класс пользователей 2
 - ...
 - 3.2.m Класс пользователей m
- 3.3 Требования к рабочим характеристикам
- 3.4 Проектные ограничения
- 3.5 Атрибуты системы программного обеспечения
- 3.6 Другие требования

Шаблон раздела 3 SRS (по объектам)

- 3.1 Внешние интерфейсы
 - 3.1.1 Интерфейсы пользователя
 - 3.1.2 Аппаратные интерфейсы
 - 3.1.3 Интерфейсы программного обеспечения
 - 3.1.4 Интерфейсы связи
- 3.2 Классы/объекты
 - 3.2.1 Класс/объект 1
 - 3.2.1.1 Атрибуты
 - 3.2.1.1.1 Атрибут 1
 - ...
 - 3.2.1.1.n Атрибут n
 - 3.2.1.2 Функции
 - 3.2.1.2.1 Функциональное требование 1.1
 - ...
 - 3.2.1.2.m Функциональное требование 1.m
 - ...
 - 3.2.1.3 Сообщения
 - ...
 - 3.2.k Класс/объект k
 - 3.3 Требования к рабочим характеристикам
 - 3.4 Проектные ограничения
 - 3.5 Атрибуты системы программного обеспечения
 - 3.6 Другие требования

Шаблон раздела 3 SRS (по свойствам)

- 3.1 Внешние интерфейсы
 - 3.1.1 Интерфейсы пользователя
 - 3.1.2 Аппаратные интерфейсы
 - 3.1.3 Интерфейсы программного обеспечения

- 3.1.4 Интерфейсы связи
- 3.2 Свойства системы
 - 3.2.1 Свойства системы 1
 - 3.2.1.1 Назначение свойства
 - 3.2.1.2 Последовательность стимулов/откликов
 - 3.2.1.3 Ассоциированные функциональные требования
 - 3.2.1.3.1 Функциональные требования 1
 - ...
 - 3.2.1.3.n Функциональные требования n
 - 3.2.2 Свойство системы 2
 - ...
 - 3.2.m Свойство системы m
- 3.3 Требования к рабочим характеристикам
- 3.4 Проектные ограничения
- 3.5 Атрибуты системы программного обеспечения
- 3.6 Другие требования

Шаблон раздела 3 SRS (по функциональной иерархии)

- 3.1 Внешние интерфейсы
 - 3.1.1 Интерфейсы пользователя
 - 3.1.2 Аппаратные интерфейсы
 - 3.1.3 Интерфейсы программного обеспечения
 - 3.1.4 Интерфейсы связи
- 3.2 Функциональные требования
 - 3.2.1 Информационные потоки
 - 3.2.1.1 Схема потока данных 1
 - ...
 - 3.2.1. n Схема потока данных n
 - 3.2.2 Описание процессов
 - 3.2.2.1 Описание процесса 1
 - ...
 - 3.2.2.m Описание процесса m
 - 3.2.3 Спецификации структуры данных
 - 3.2.3.1 Спецификация структуры 1
 - ...
 - 3.2.3.k Спецификация структуры k
 - 3.2.4 Словарь данных
 - 3.2.4.1 Элемент данных 1
 - ...
 - 3.2.4. k Элемент данных k
- 3.3 Требования к рабочим характеристикам
- 3.4 Проектные ограничения

3.5 Атрибуты системы программного обеспечения

3.6 Другие требования

Шаблон раздела 3 SRS (множественная организация)

3.1 Внешние интерфейсы

3.1.1 Интерфейсы пользователя

3.1.2 Аппаратные интерфейсы

3.1.3 Интерфейсы программного обеспечения

3.1.4 Интерфейсы связи

3.2 Функциональные требования

3.2.1 Класс пользователей 1

3.2.1.1 Свойство 1.1

3.2.1.1.1 Назначение свойства

3.2.1.1.2 Последовательность стимулов/откликов

3.2.1.1.3 Ассоциированные функциональные требования

3.2.1.1.3.1 Функциональные требования 1

...

3.2.1.1.3.k Функциональные требования k

...

3.2.1.n Свойство 1.n

3.2.2 Класс пользователей 2

...

3.2.m Класс пользователей m

3.3 Требования к рабочим характеристикам

3.4 Проектные ограничения

3.5 Атрибуты системы программного обеспечения

3.6 Другие требования

1.4 Задания для лабораторной работы 1

Задание 1. Ознакомиться с особенностями рабочего интерфейса программы IBM Rational Rose и изучить особенности разработки диаграммы вариантов использования и редактирования свойств ее элементов. Продублировать построение такой диаграммы средствами IBM Rational Rose на примере обслуживания клиента банкоматом (пункт 1.2).

Задание 2. Изучить спецификации требований и рекомендации по написанию эффективных вариантов использования.

Задание 3. Разработать диаграммы вариантов использования, соответствующие спецификации требований и сценарии (по аналогии с задачей обслуживания клиента банкоматом из пункта 1.2)

применительно к собственной проблемной тематике. При выборе проблемной тематики рекомендуется ориентироваться на будущую дипломную работу. В случае затруднения с выбором темы, ниже предлагается список из возможных вариантов. Для создания спецификаций требований воспользуйтесь шаблонами SRS. При описании интерфейсов и функций соответствующие тексты проиллюстрируйте диаграммами вариантов использования. Для общей части (разделы 1 и 2) это будет обобщенная диаграмма, состоящая из крупных блоков. При оформлении раздела 3 «Специфические требования» необходимо использовать как минимум два шаблона SRS (из семи приведенных), которые в наибольшей степени подходят под ваш проект. Каждый вариант шаблона раздела 3 проиллюстрируйте набором диаграмм вариантов использования, расшифровывающих соответствующие блоки обобщенной диаграммы.

Задание 4. Оформить отчет.

Контрольные вопросы

1. Какое основное назначение диаграмм вариантов использования в бизнес-проектировании или разработки проекта программного обеспечения?

2. Что такое отношение ассоциации и для чего оно предназначено в диаграмме вариантов использования?

3. Для чего используется навигация ассоциации и как ее можно изменять?

4. В чем состоит отличие отношений зависимости «include» от «extend»?

5. С помощью каких двух основных составляющих специфицируется функциональное требование?

6. Чем отличаются функциональные и нефункциональные требования?

7. Опишите основные элементы шаблона сценария вариантов использования.

8. Опишите общую схему типичного хода событий в сценарии выполнения вариантов использования.

Рекомендуемый список практических задач для выполнения лабораторной работы.

1. Проектирование программного средства автоматизации составления расписания уроков в школе.

2. Проектирование программного средства описания истории болезни.

3. Проектирование программного средства автоматизации описания результатов УЗИ-диагностики.

4. Проектирование программного средства автоматизации описания результатов диагностики по Фоллю.
5. Проектирование программного средства описания истории содержания коров молочной фермы.
6. Проектирование программного средства описания истории возделывания сельскохозяйственных культур на полях сельскохозяйственного предприятия.
7. Проектирование программного средства описания истории содержания лесных участков леспромхозом.
8. Проектирование программного средства описания истории содержания жилого многоэтажного дома жилищно-коммунальной службой.
9. Проектирование программного средства автоматизации комплексного обслуживания жителей удаленного сельского населенного пункта.
10. Проектирование программного средства автоматизации формирования заказов в системе школьного питания.
11. Проектирование программного средства для составления справочника движения городского транспорта.
12. Проектирование программного средства для составления справочника спроса и предложения товаров крупного торгового центра.
13. Проектирование программного средства для составления справочника спроса и предложения услуг по ремонту квартир.
14. Проектирование программного средства автоматизации функционирования тепличного хозяйства на основе гидропоники.
15. Проектирование программного средства автоматизации функционирования домашней отопительной системы.
16. Проектирование программного средства системы дешифрования.
17. Проектирование программного средства управления движением поездов.
18. Проектирование программного средства управления дислокационными операциями в морском порту.
19. Проектирование программного средства управления отраслью заготовок.
20. Проектирование программного средства управления обжигом цементного клинкера.
21. Проектирование программного средства управления в энергосистеме.
22. Проектирование программного средства планирования действия робота.

23. Проектирование программного средства описания истории обслуживания персонального компьютера.
24. Проектирование системного проекта паевого фонда.
25. Проектирование программного средства автоматизации деятельности автобазы, занимающейся перевозками пород горно-обогатительного комбината.
26. Проектирование программного приложения учета материальных ценностей.
27. Проектирование программного приложения учета договоров с поставщиками.
28. Проектирование web-сайта по проведению конференций.
29. Проектирование программного средство учета текущей успеваемости студентов.
30. Проектирование web-сайта по организации транспортных маршрутов.
31. Проектирование web-сайта по учету публикаций.
32. Проектирование web-сайта для регистрации участников конференции.
33. Проектирование web-сайта кафедры.
34. Проектирование программного средства регистрации и поиска автомашин.
35. Проектирование программного средства автоматизации учёта ассортимента обуви.
36. Проектирование программного средства автоматизации библиотечного каталога.
37. Проектирование программного средства автоматизации олимпиады по программированию.
38. Проектирование программного средства автоматизации учёта студентов.
39. Проектирование программного средства для учета сдачи спортивных нормативов.
40. Проектирование программного средства автоматизации учёта сведений о футбольных командах.
41. Проектирование приложения средства учета парка общественного транспорта.
42. Проектирование программного приложения автоматизации расписания факультета.
43. Проектирования приложения автоматизация ведомости на выдачу стипендии.
44. Проектирование приложения для сбора и поиска информации о представителях лесной растительности.

45. Проектирование веб-сайта о курсах иностранных валют в определенные сутки.
46. Проектирование приложения для сбора и поиска информации о преподавателях кафедры.
47. Проектирование приложения для сбора и поиска информации о структуре университета.
48. Проектирование приложения для сбора и поиска информации о факультетах университета.
49. Проектирование приложения для учета стипендии студентов.
50. Проектирование веб-сайта фирмы, реализующей компьютерные комплектующие.
51. Проектирование приложения для учета информации о телевизорах.
52. Проектирование приложения учета товара в магазине бытовой техники
53. Проектирование приложения для регистрации дефектов и отказов оборудования.
54. Проектирование веб-сайта о оптических носителях информации.
55. Проектирование приложение учета товара в магазине множительной и копировальной техники.
56. Проектирование интернет-магазина мобильных телефонов.

2 Лабораторная работа №2 Управление разработкой ПО в среде Microsoft Project

2.1 Работа в среде Microsoft Project

Предлагаемые Microsoft инструменты управления проектами предусматривают наличие одного из вариантов конфигурации MS Project, в частности, MS Project 2007 [8-10].

Необходимо отметить, что все программные инструменты управления проектами (в том числе и MS Project) следует рассматривать и использовать как средства поддержки принятия решений менеджером проекта: с помощью MS Project менеджер может оценить эффективность нескольких альтернативных вариантов реализации проекта и выбрать стратегию, в наибольшей степени отвечающую интересам компании и целям проекта.

Применение MS Project на стадии планирования поможет руководителю ответить на следующие вопросы:

- насколько вообще реально воплощение в жизнь данного проекта?
- какие конкретно работы необходимо выполнить для достижения целей проекта?
- какой состав исполнителей, соисполнителей и какие виды материальных ресурсов потребуются для реализации проекта?
- какова стоимость проекта и как наиболее выгодно распределить во времени финансовые затраты на реализацию проекта? Кто должен отвечать за те или иные виды работ?
- насколько велик риск и каков возможный ущерб при завершении проекта на той или иной стадии?

Для ответа на первый вопрос требуется провести полный анализ проекта по методу критического пути с использованием ресурсного планирования, однако без излишней детализации. В этом отношении весьма большую помощь могут оказать шаблоны, входящие в состав стандартной конфигурации MS Project. Каждый из шаблонов относится к определенной сфере, и может считаться своеобразным стандартом соответствующего плана проекта. Внося в него необходимые коррективы в соответствии с особенностями конкретного проекта, можно получить вполне реалистичную оценку возможного развития событий и требуемых затрат. Например, на рисунке 2.1 приведен фрагмент диаграммы Ганта.

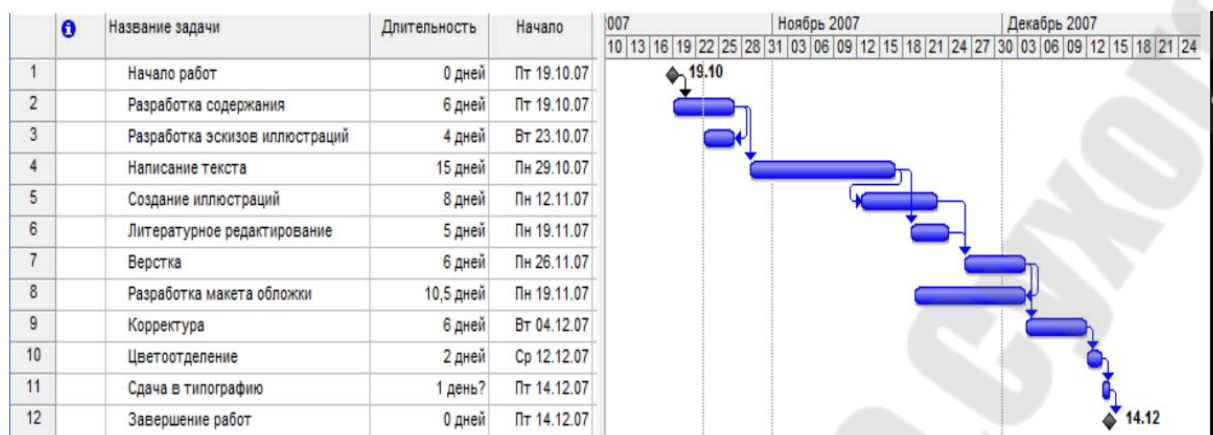


Рисунок 2.1 – Диаграмма Ганта

Ответ на второй вопрос также может быть получен с помощью одного из стандартных расписаний. Если структуру проекта приходится создавать вручную (нет шаблона 0, то MS Project 2007 способен оказать существенную помощь благодаря средствам построения сетевого графика (Network Diagram). Для работ проекта автоматически устанавливаются параметры, заданные по умолчанию (такие как длительность, календарные даты начала и окончания и т. д.). На рисунке 2.2 показан один из возможных вариантов представления сетевого графика, наиболее близкий его «бумажному» аналогу.

На основе сетевого графика автоматически формируется календарный план в виде диаграммы Ганта. Определив структуру расписания в виде сетевого графика, вы получаете «заготовку» календарного графика с привязкой сроков выполнения работ к реальным датам (рисунок 2.3).

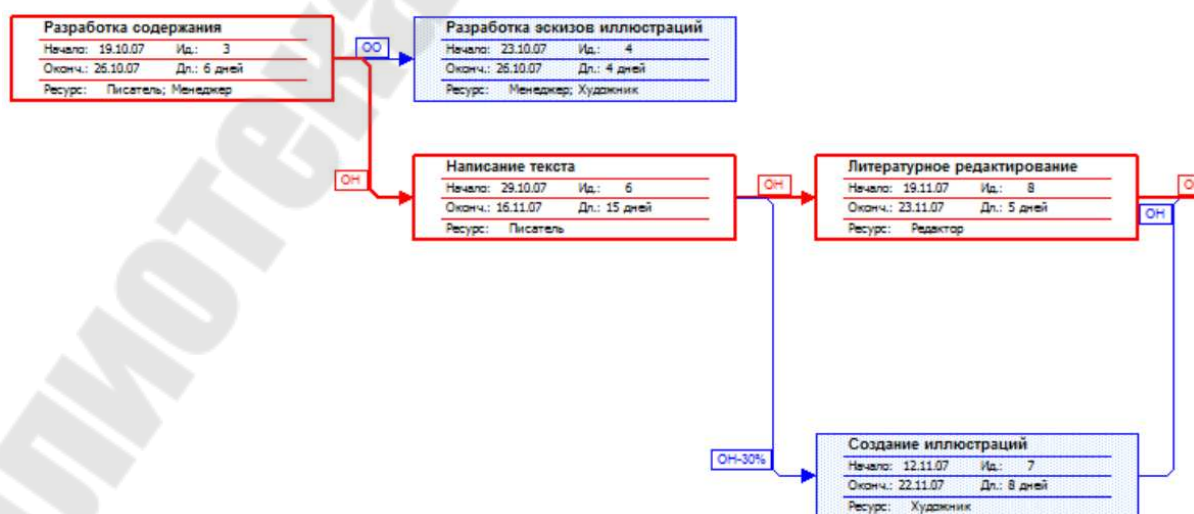


Рисунок 2.2 – Сетевой график

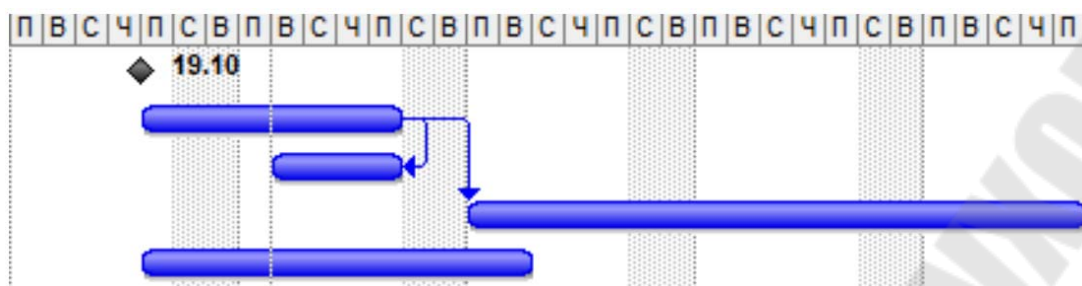


Рисунок 2.3 – Разные типы связей между задачами

Чтобы получить ответ на третий по счету из перечисленных выше вопросов, требуется выполнить назначение ресурсов. В качестве ресурсов проекта могут быть заданы либо уникальные для него исполнители и материалы, либо назначены виды ресурсов, использовавшихся в предыдущих проектах (или взятые из шаблонов). Обобщенную информацию об используемых в проекте ресурсах можно получить с помощью таблицы ресурсов (рисунок 2.4), а более детальную информацию – на основе анализа назначений.

	И	Название ресурса	Тип	Единицы измерения материалов	Краткое название	Группа	Макс. единиц	Стандартная ставка	Ставка сверхурочных	Затраты на исполыз.	Начисление	Базовый календарь
1		Писатель	Трудовой		П	Люди	100%	0,00р./ч	0,00р./ч	7 000,00р.	По окончании	Стандартный
2		Редактор	Трудовой		Р	Люди	100%	50,00р./ч	100,00р./ч	0,00р.	Пропорционалы	Стандартный
3		Художник	Трудовой		Х	Люди	100%	70,00р./ч	140,00р./ч	0,00р.	Пропорционалы	Стандартный
4		Верстальщик	Трудовой		В	Люди	100%	50,00р./ч	100,00р./ч	0,00р.	Пропорционалы	Стандартный
5		Корректор	Трудовой		К	Люди	100%	50,00р./ч	100,00р./ч	0,00р.	Пропорционалы	Стандартный
6		Менеджер	Трудовой		М	Люди	100%	100,00р./ч	200,00р./ч	0,00р.	Пропорционалы	Стандартный
7		Компьютер	Трудовой		К	Техника	400%	0,00р./ч	0,00р./ч	0,00р.	Пропорционалы	Стандартный

Рисунок 2.4 – Таблица ресурсов

Для каждого ресурса могут быть построены гистограммы его загрузки и стоимости. После назначения очередного ресурса (с указанием его стоимости и объема) выполняется автоматический пересчет стоимости проекта, благодаря чему очень легко получить сравнительную оценку различных вариантов назначений.

Для проведения стоимостного анализа проекта в MS Project 2007 используется так называемый «метод освоенного объема» (Earned Value Analysis), с помощью которого может быть проведен анализ затрат либо на текущую дату, либо на заданную календарную дату.

Любой, даже самый хороший план не застрахован от случайностей.

Чтобы адекватно анализировать риски проекта, необходимо иметь его детализированный план. Как правило, выбор методов и средств для анализа рисков зависит от специфики проекта, состава и

уровня подготовки группы проекта. В частности, простым и вместе с тем эффективным средством является сравнение нескольких версий (сценариев) расписания проекта.

При использовании **анализа по методу PERT** таких сценариев должно быть три: **наиболее вероятный (ожидаемый), оптимистичный и пессимистичный**. Для сравнительной оценки длительности проекта по этим трем сценариям в составе MS Project 2007 имеется специальный инструмент – процедура *PERT*, с ее помощью вы можете описать и сравнить между собой параметры проекта для трех альтернативных сценариев развития событий.

Для более сложных ситуаций могут быть созданы соответствующие макросы, реализованные с помощью VBA (язык программирования Visual Basic Application).

Завершая короткий обзор основных возможностей MS Project 2007, отметим, что на любой стадии работы над проектом вы всегда будете чувствовать поддержку со стороны разработчиков. Либо в виде всплывающих окон с подсказками, либо в форме смарт-тегов, либо в какой-то другой форме.

2.2 Задания для лабораторной работы 2

Задание 1. Создать проект *Внедрение бухгалтерской системы*, предназначенный для автоматизации бухгалтерии небольшого предприятия, состоящей из 10 человек. Дата начала проекта – 1 ноября 2010 года. Перечень задач проекта, их связи и длительности приведены в таблице 1.

Таблица 2.1 – Перечень задач проекта, их связи и длительности

№	Название задачи	Длит (дн)	Предшественники
1	Начало проекта	0	
2	Выбор системы		
3	Изучение рынка бухгалтерских систем	7	1
4	Составление требований к бухгалтерским системам	7	1
5	Консультации с фирмами-разработчиками	7	3;4
6	Принятие окончательного решения	2	5
7	Выбор завершен	0	6
8	Приобретение программного обеспечения		
9	Заключение договоров	6	2
10	Оплата за ПО	2	9
11	Оформление ПО на баланс	3	10
12	Приобретение ПО завершено	0	11
13	Составление проекта сети		
14	Разработка архитектуры сети	7	7

15	Проработка физического размещения сети	5	14
16	Проект сети завершен	0	15
17	Приобретение компьютеров и сетевого оборудования		
18	Сбор информации о поставщиках и предложениях	7	7
19	Анализ и выбор поставщика	5	14;18
20	Заключение договоров	5	19
21	Оплата за оборудование	2	20
22	Оформление оборудования на баланс	3	21
23	Приобретение оборудования завершено	0	22
24	Обучение администратора и программиста		
25	Курсы администраторов	18	16
26	Курсы программистов	18	12
27	Сдача сертификационных экзаменов	3	25;26
28	Обучение завершено	0	27
29	Монтаж локальной сети		
30	Установка компьютеров на рабочих местах	3	23;28
31	Монтаж кабеля	10	23;28
32	Монтаж сетевых устройств	10	23;28
33	Подключение кабеля к компьютерам и сетевым устройствам	5	30;31;32
34	Монтаж завершен	0	33
35	Установка ПО на компьютеры		
36	Установка сервера	5	34
37	Создание доменов и пользователей	7	36
38	Проверка и настройка работы сети	5	37
39	Настройка сети завершена	0	38
40	Ввод начальных данных		
41	Ввод справочников	40	39
42	Ввод начальных остатков	40	41
43	Ввод начальных данных завершен	0	42
44	Обучение персонала		
45	Принципы работы системы	3	39
46	Изучение интерфейса	5	45
47	Изучение справочников	20	41;46
48	Изучение документов и журналов	30	42;47
49	Обучение завершено	0	48
50	Передача в эксплуатацию		
51	Формирование тестовой отчетности	5	49
52	Акт ввода в эксплуатацию	3	51
53	Передача в эксплуатацию завершена	0	52
54	Конец проекта	0	53

Фазы в проекте выделены полужирным курсивом, а вехи имеют нулевую длину. Названия задач, входящих в фазу, выделены отступом слева.

В полученной таблице произведите следующие операции:

- между задачами 10 и 11 установить задержку в 5 дней, необходимую для прохождения безналичной оплаты;

- между задачами 21 и 22 установить задержку в 7 дней, необходимую для прохождения безналичной оплаты и доставки оборудования;
- установить тип связи между задачами 41 и 47 начало-начало и задержку в 5 дней;
- установить ограничение для задачи 42 по дате выполнения – не ранее 1.05.2011.

Задание 2. При выполнении задания используются варианты проектов, созданных при выполнении предыдущей работы.

1. Выполнить форматирование таблицы ввода диаграммы Ганта ранее созданного проекта *Внедрение бухгалтерской системы* с помощью следующих операций:

- удалить столбец идентификаторов;
- добавить столбец *Критическая задача*;
- назначить стили текста, выделив разными цветами фазы, вехи, критические и некритические задачи. При помощи стилей текста установить для заголовков строк и столбцов жирный шрифт коричневого цвета, а для среднего уровня шкалы времени сиреневый цвет;
- сохранить изменения в файле *Бух2_1.mpp*.

2. В первоначальном файле проекта выполнить сортировку задач таблицы по дате начала, а затем по дате окончания. Результат сохранить в файле *Бух2_2.mpp*.

3. В первоначальном файле проекта выполнить многоуровневую сортировку таблицы сначала по возрастанию признака критической задачи, а затем по убыванию даты окончания. Результат сохранить в файле *Бух2_3.mpp*.

4. В первоначальном файле проекта используя структурный фильтр отобразить только задачи 1 уровня. Результат сохранить в файле *Бух2_4.mpp*.

5. В первоначальном файле проекта используя автофильтр отобразить задачи, которые начинаются в следующем месяце и имеют длительность больше 15 дней. Результат сохранить в файле *Бух2_5.mpp*.

6. В первоначальном файле проекта используя предопределенный фильтр отобразить только суммарные задачи. Результат сохранить в файле *Бух2_6.mpp*.

7. В первоначальном файле проекта при помощи предопределенной группировки сгруппировать отдельно вехи проекта и задачи, не являющиеся вехами. Результат сохранить в файле *Бух2_8.mpp*.

8. В первоначальном файле проекта создать определенную пользователем группировку и поместить ее в системное меню. На первом уровне задачи группируются на критические и некритические по убыванию параметра Критическая задача. На втором уровне задачи группируются по

возрастанию длительности. Применить созданную группировку. Результат сохранить в файле *Бух2_9.mpp*.

9. В первоначальном файле проекта создать временную группировку задач по интервалам длительности. Использовать недельный и месячный интервалы. Применить группировку. Результат сохранить в файле *Бух2_10.mpp*.

Задание 3. Создать проект для своей проблемной задачи (из лабораторной работы 1) по примеру задания 1 и построить соответствующую диаграмму Ганта. Распланировать сроки выполнения задачи, ориентируясь на дату защиты дипломного проекта.

Задание 4. Для ранее созданного проекта *Внедрение бухгалтерской системы* создать список ресурсов в соответствии с параметрами, перечисленными в таблице 2.2, а также назначения ресурсов в соответствии с таблицей 2.3.

Таблица 2.2 – Список ресурсов

Наименование	Тип	Таблица норм	Станд. ставка	Ставка сверхур.	Затраты на исп.
Главбух	Т	АВ	90000р./мес	500р./ч	30000р
Администратор	Т	АВ	70000р./мес	450р./ч	40000р
Программист	Т	АВ	60000р./мес	400р./ч	50000р
Техник	Т	А	40000р./мес	250р./ч	-
Расчетчик1	Т	А	40000р./мес	250р./ч	-
Расчетчик2	Т	А	40000р./мес	250р./ч	-
Расчетчик3	Т	А	40000р./мес	250р./ч	-
Бухгалтер мат. учета1	Т	А	40000р./мес	250р./ч	-
Бухгалтер мат. учета2	Т	А	40000р./мес	250р./ч	-
Бухгалтер учета ОС и НМА	Т	А	40000р./мес	250р./ч	-
Бухгалтер учета ОС	Т	А	40000р./мес	250р./ч	-
Бухгалтер учета реализации	Т	А	40000р./мес	250р./ч	-
Бухгалтер производственного учета	Т	А	40000р./мес	250р./ч	-
Компьютер	М	А	15000		-
Сервер	М	А	50000		-
Принтер	М	А	5000		-
МФУ	М	А	7000		-
Сетевой кабель	М	А	-		15000
Сетевой концентратор	М	А	3000		-
Панель	М	А			10000
Разъемы и розетки	М	А	-		15000
Бухгалтерская система	М	А	-		100000
Офисный пакет	М	А	-		70000
ОС рабочей станции	М	А	-		60000
Серверная ОС	М	А	-		30000
DVD-матрица	М	А	10		-

Интернет	3			
Междугородние переговоры	3			
Оплата курсов	3			

Таблица 2.3 – Назначение ресурсов для задач

Задача	Ресурс	Ед., (затраты)	Табл. норм затрат
Изучение рынка бухгалтерских систем	Администратор Интернет	100	А 1500
Составление требований к бухгалтерским системам	Администратор Главбух	100 20	А А
Консультации с фирмами-разработчиками	Администратор Междугородние переговоры Интернет	100	А 2000 1000
Принятие окончательного решения	Администратор Главбух	100 100	А А
Заключение договоров	Администратор Программист Главбух	100 100 100	А А А
Оплата за ПО	Главбух Бухгалтерская система Офисный пакет ОС рабочей станции Серверная ОС	10	А А А А А
Оформление ПО на баланс	Бухгалтер учета ОС и НМА	30	А
Разработка архитектуры сети	Администратор Программист Техник	100 100 50	А А А
Проработка физического размещения сети	Администратор Программист Техник	100 100 100	А А А
Сбор информации о поставщиках и предложениях	Администратор Интернет Междугородние переговоры	50	А 1000 1500
Анализ и выбор поставщика	Администратор Главбух Интернет	50 20	А А 1000
Заключение договоров	Администратор Главбух	100 50	А А
Оплата за оборудование	Главбух Компьютер Сервер Принтер МФУ Сетевой кабель Сетевой концентратор Панель Разъемы и розетки	30 12 1 2 2 2	А А А А А А А А А

Оформление оборудования на баланс	Бухгалтер учета ОС	70	A
Курсы администраторов	Администратор Оплата курсов	100	A 25000
Курсы программистов	Программист Оплата курсов	100	A 2000
Сдача сертификационных экзаменов	Администратор Программист	100 100	A A
Установка компьютеров на рабочих местах	Техник	100	A
Монтаж кабеля	Техник	100	A
Монтаж сетевых устройств	Техник	100	A
Подключение кабеля к компьютерам и сетевым устройствам	Техник	100	A
Установка сервера	Администратор	100	A
Создание доменов и пользователей	Администратор	100	A
Проверка и настройка работы сети	Администратор Программист	100 100	A A
Ввод справочников	Администратор Программист Расчетчик1 Расчетчик2 Расчетчик3 Бухгалтер мат. учета1 Бухгалтер мат. учета2 Бухгалтер учета ОС и НМА Бухгалтер учета ОС Бухгалтер учета реализации Бухгалтер производ. учета DVD-матрица	100 100 30 30 30 50 50 50 50 50 50 50 10	B B A A A A A A A A A A
Ввод начальных остатков	Администратор Программист Главбух DVD-матрица	100 100 50 10	B B A A
Принципы работы системы	Администратор Главбух Расчетчик1 Расчетчик2 Расчетчик3 Бухгалтер мат. учета1 Бухгалтер мат. учета2 Бухгалтер учета ОС и НМА Бухгалтер учета ОС Бухгалтер учета реализации Бухгалтер производ. учета	50 50 50 50 50 50 50 50 50 50 50	A A A A A A A A A A A
Изучение интерфейса	Программист Главбух Расчетчик1 Расчетчик2	50 50 50 50	A A A A

	Расчетчик3	50	A
	Бухгалтер мат. учета1	50	A
	Бухгалтер мат. учета2	50	A
	Бухгалтер учета ОС и НМА	50	A
	Бухгалтер учета ОС	50	A
	Бухгалтер учета реализации	50	A
	Бухгалтер производственного учета	50	A
Изучение справочников	Программист	50	A
	Главбух	50	A
	Расчетчик1	50	A
	Расчетчик2	50	A
	Расчетчик3	50	A
	Бухгалтер мат. учета1	50	A
	Бухгалтер мат. учета2	50	A
	Бухгалтер учета ОС и НМА	50	A
	Бухгалтер учета ОС	50	A
	Бухгалтер учета реализации	50	A
	Бухгалтер производ. учета	50	A
Изучение документов и журналов	Программист	50	A
	Главбух	50	A
	Расчетчик1	50	A
	Расчетчик2	50	A
	Расчетчик3	50	A
	Бухгалтер мат. учета1	50	A
	Бухгалтер мат. учета2	50	A
	Бухгалтер учета ОС и НМА	50	A
	Бухгалтер учета ОС	50	A
	Бухгалтер учета реализации	50	A
	Бухгалтер производс. учета	50	A
Формирование тестовой отчетности	Администратор	100	A
	Программист	100	A
	Главбух	100	A
Акт ввода в эксплуатацию	Администратор	50	A
	Главбух	50	A

Контрольные вопросы

1. На какие вопросы планирования может помочь ответить модель, созданная в Microsoft Project?
2. Что такое сетевой график?
3. Какие типы связей между задачами существуют в Microsoft Project?
4. Какие виды ресурсов существуют в Microsoft Project?
5. Что такое анализ по методу PERT?
6. Что такое фазы, вехи и завершающие задачи в Microsoft Project?
7. Что такое назначения в Microsoft Project и какие задачи планирования они решают?

Список использованных источников

1. Милошевич Д. Набор инструментов для управления проектами. – М.: Компания АиТи; ДМК Пресс, 2008. – 729 с.
2. Шафер Д., Фатрел Р., Шафер Л. Управление программными проектами: достижение оптимального качества при минимуме затрат. – М.: Вильямс, 2003. – 1136 с.
3. Хелдман К. Профессиональное управление проектом – М.: БИНОМ. Лаборатория знаний, 2005. – 517 с.
4. Ройс У. Управление проектами по созданию программного обеспечения. – М.: Издательство «Лори», 2000. – 431 с.
5. Брауде Э. Технологии разработки программного обеспечения. – СПб.: Питер, 2004. – 655 с.
6. Беркун С. Искусство управления IT-проектами. – СПб.: Питер, 2007. – 400 с.
7. Леоненков, А. Визуальное моделирование в среде IBM Rational A.Rose 2003 / А. Леоненков. [Электронный ресурс]. – 2016. – Режим доступа: <http://www.intuit.ru/studies/courses/14/14/info>. – Дата доступа : 15.12.2016.
8. Microsoft Office Project 2007. Библия пользователя. М.: Вильямс, 2008. – 800 с.
9. Microsoft Office Project Professional 2007. Управление проектами: Практическое пособие. – СПб.: Корона-Век, 2008. – 480 с.
10. Богданов В. Управление проектами в Microsoft Office Project 2007. Учебный курс. – СПб.: Питер, 2008. – 592 с.

Осипенко Александр Николаевич

**УПРАВЛЕНИЕ РАЗРАБОТКОЙ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**Практикум
по одноименному курсу
для слушателей специальности переподготовки
1-40 01 73 «Программное обеспечение
информационных систем»
заочной формы обучения**

Подписано к размещению в электронную библиотеку
ГГТУ им. П. О. Сухого в качестве электронного
учебно-методического документа 27.10.17.

Пер. № 108Е.

<http://www.gstu.by>