

Методические указания к лабораторным работам по дисциплине «Разработка и анализ требований проектирования программного обеспечения»

Лабораторная работа № 1. Формирование списка требований к типовому программному продукту

Цель работы: познакомиться с понятием «требование к программному продукту», научиться формулировать требования с учетом ожиданий пользователей от программного продукта, систематизировать требования, анализировать на предмет дублирования, логической зависимости, двусмысленности, возможности реализации, соответствия законодательству .

Задание

1. Выбрать знакомую вам категорию программных продуктов. Рекомендуемый список продуктов приведен в Приложении 1. Также можно использовать программные продукты, не включенные в список, но по согласованию с преподавателем.

2. Сформировать таблицу для фиксации требований следующего содержания:

Идентификатор требования	Требование	Категория требования	Родительское требование (№)	Дочернее требование (№)	Статус требования	Приоритет требования

Рекомендация: для работы используйте электронные таблицы.

3. На основе личного опыта работы с подобными программными продуктами сформулировать не менее 20 различных требований к программному продукту и зафиксировать их в таблице в столбце Требование. Фиксируются только атомарные требования, т. е. требование не должно включать в себя два и более требований. Требования идентифицировать любым способом (например, 1,2,3 ...). Идентификаторы в процессе работы не изменять.

4. Определить категорию требования. Перечень используемых категорий будет зависеть от категории рассматриваемого программного продукта. Возможны следующие универсальные

категории: требования к интерфейсу пользователя, настройкам, системному окружению, коммуникационным возможностям, форматам данных.

5. Заполнить поле Статус требования. Возможные формулировки статуса: предложено , одобрено, отклонено (для требований, которые в данный момент не будут реализованы, но в дальнейшем они, возможно, будут включены в рассмотрение), удалено (для требований, которые никогда не будут реализованы или дублируют другие требования) , реализовано (в данной работе не использовать, т.к. реализация требований не предусматривается). Требования со статусом «удалено» не удалять из таблицы!

6. Проследить зависимости требований и показать их в таблице в столбцах Родительское требование и Дочернее требование.

7. С учетом связей требований расставить приоритеты.

8. Результаты показать преподавателю.

Приложение 1. Список категорий программных продуктов

- a) Текстовый редактор.
- b) Браузер.
- c) Система управления версиями.
- d) Графический редактор.
- e) Электронные таблицы.
- f) Программа для просмотра больших текстов.
- g) Настольная издательская система.
- h) Инструментальная среда разработки.
- i) Программа для математического моделирования.
- j) Программа для доступа к библиотечному каталогу.
- k) Система indoor-навигации.
- l) Система "умный дом".
- m) Личный кабинет на www.sut.ru.
- n) Система доступа к учебным материалам.

Лабораторная работа № 2. Обзор конкурентов программного продукта

Цель работы: познакомиться с методикой анализа конкурирующих продуктов и технологий для оценки новизны и обоснования актуальности продукта, предлагаемого для разработки.

Задание

1. Для категории продуктов, рассмотренных в лабораторной работе № 1, подобрать 3-4 программных продукта данной категории.

2. Для данной категории программных продуктов сформировать перечень требований к функциям и качеству. Количество требований — не менее 6.

3. Провести анализ выбранных продуктов, реализованы ли в них данные функции, а результаты анализа представить в виде таблицы. Пример таблицы — в Приложении 1. Источником информации о продукте может быть личный опыт, документация на программный продукт, статьи-обзоры программных продуктов. В таблице следует отметить следующее: функция реализована полностью, частично или с некоторыми условиями, не реализована, нет сведений о реализации.

4. Сформулировать и обосновать предложение о составе функций и качественных характеристиках продукта, предлагаемого для разработки.

5. Результаты анализа и обоснование новизны и актуальности предлагаемого продукта представить преподавателю.

Приложение 1

Таблица 1. Сравнительная характеристика простейших текстовых редакторов

Текстовый редактор	ОС помимо Windows	Возможности								Кодировки			
		Работа без устанавки	Открытие нескольких документов в одном окне	HEX-редактор	Нумерация строк	Смена конца строк (CRLF)	Подсветка синтаксиса	Быстрая вставка даты и времени	Замена текста на основе регулярных выражений	Win1251	DOS866	KOI8-R	Unicode
Стандартный блокнот	Нет	Часть ОС	Нет	Нет	Нет	Нет	Нет	Да	Нет	Да	Шаблон: Да	Нет	Да

<u>Windows</u>										шриф т termi nal			
<u>Notepad++</u>	Нет	Да	Да	Да	Да	Да	Да	Да	Да	Да	Да	Да	
<u>Notepad2</u>	Нет	Да			Да	Да	Да			Да		Да	
<u>EditPlus</u>	Нет		Да		Да	Да	Да	Да	Да	Да	Да	Да	
<u>UltraEdit</u>	Нет		Да	Да			Да	Да	Да	Да		Да	
<u>CrimsonEditor</u>	Нет		Да				Да	Да	Да	Да	Нет	Нет	Да
<u>TextPad</u>	Нет		Да	Да			Да	Нет	Да	Да	Да	Нет	Да
<u>RJ TextEd</u>	Нет	Да	Да	Да	Да	Да	Да	Да	Да	Да	Да	Да	Да
<u>AkelPad</u>	Нет	Да	Да	Нет	Да(п лагин)	Да	Да(пл агин)	Да	Да(плагин)	Да	Да	Да	Да
<u>Boxer TextEditor</u>	Нет[6]		Да	Да			Да		Да	Да	Да	Нет	Нет
<u>jEdit</u>	Да		Да		Да		Да		Да	Да	Да	Да	Да
<u>DPAD</u>	Нет	Да	Да	Нет	Да	Да	Да	Да	Да	Да	Да	Да	Да

Замечание. В таблице реализованные функции отмечены словом «да», нереализованные - «нет». Функции, для реализованных частично добавлен комментарий. Пустые ячейки таблицы означают, что о реализации данной функции сведений нет.

Лабораторная работа № 3. Разработка презентации для индивидуального проекта программного продукта

Цель работы: научиться готовить предложение по разработке программного продукта для представления на конкурс, инвестору, руководству.

Задание

1. Сформулировать устно идею, основные функции предлагаемого программного продукта.
2. Подготовить обзор конкурентов (аналогов) предлагаемого программного продукта.

3. Сформировать перечень основных функций программного продукта. К основным функциям можно отнести функции, приносящие пользу потребителям и представляющие некоторый алгоритм взаимодействия пользователя и программного продукта. Основная функция в дальнейшем может быть разбита на подфункции.

4. Подготовить презентацию, включающую следующий набор слайдов:

- титульный лист (название проекта, автор, год),
- исходные стимулы для разработки (краткое описание проблемы),
- назначение и краткое описание продукта, целевая аудитория,
- отличия от конкурентов,
- бизнес-цели и критерии успеха реализации,
- бизнес-риски,
- основные функции,
- особенности развертывания (системные требования, организация рекламной кампании и обучения заинтересованных лиц).

Также в презентацию могут быть включены обзор конкурентов, рисунки, скриншоты, диаграммы, иллюстрирующие актуальность и функции продукта.

Лабораторная работа № 4. Разработка прототипа программного продукта

Цель работы: познакомиться с понятием прототипа программного продукта, задачами, решаемыми с помощью прототипирования, научиться готовить горизонтальные и вертикальные прототипы.

Задание

1. Подготовить горизонтальный прототип для программного продукта, предложенного в лабораторной работе 3. Прототип представляет собой рисунок, который демонстрирует такие требования к продукту, как интерфейс пользователя, концептуальную схему данных, схему навигации по сайту. Горизонтальный прототип позволяет заказчику визуально оценить, правильно ли требования заказчика поняты разработчиком. Горизонтальный прототип может быть создан на бумаге или в любом графическом редакторе. Также интерфейс пользователя может быть представлен в виде html-кода.

2. Подготовить вертикальный прототип программного продукта. Прототип представляет собой частичную реализацию программного продукта и является исполняемым. Вертикальный

прототип демонстрирует возможность реализации алгоритма, структуры данных и запросов пользователя, возможность взаимодействия структурных компонентов программного продукта между собой и с системным окружением, предлагает оценить навигацию по сайту, удобство меню, обнаружить недостающие функции. Для реализации вертикального прототипа выбираются любые доступные студенту средства, позволяющие реализовать выбранную часть программного продукта.

Лабораторная работа № 5. Диаграмма прецедентов к проекту.

Цель работы: познакомиться с редактором UML-диаграмм, научиться представлять функции программного продукта с помощью диаграммы прецедентов (Use Case).

Задание

1. Изучить назначение диаграммы прецедентов, терминологию и обозначения основных ее элементов.

2. В графическом редакторе найти соответствующие обозначения для построения прецедентов.

3. Построить в графическом редакторе диаграмму прецедентов для программного продукта из лабораторной работы 3. На диаграмме в виде прецедента должна быть представлена каждая основная функция из лабораторной работы 3, также могут быть добавлены дополнительные прецеденты, детализирующие основные или выполняющие служебные функции.

Справочный материал

Диаграмма прецедентов — это один из типов диаграмм языка графического моделирования UML. Диаграмма прецедентов представляет основные функции программного продукта и пользователей, заинтересованных в этих функциях. Под прецедентом понимается некоторый процесс (алгоритм, функция) взаимодействия пользователя и программного продукта. Поэтому отдельный прецедент также может быть назван «вариант использования». В переводе на английский язык вариант использования преобразуется в Use Case, и именно так называется этот тип диаграммы в спецификации UML.

Основные элементы, изображаемые на диаграмме прецедентов, это прецедент и актер. Актер — это любое действующее лицо, внешнее по отношению к программному продукту. Это может быть человек, устройство или сторонний программный продукт, взаимодействующие с представляемым на диаграмме программным продуктом.

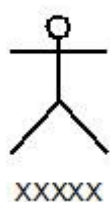
Связь между актером и прецедентом — ассоциативная. Возможна связь прецедентов на диаграмме, если прецеденты логически связаны друг с другом. Выделяют два типа связей: включение прецедента и расширение прецедента.

Включение прецедента предполагает, что включающий прецедент не может быть выполнен без выполнения включаемого прецедента.

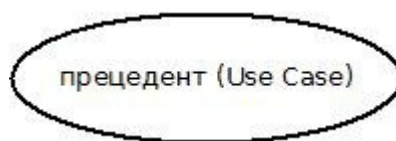
Расширение прецедента предполагает, что включающий прецедент может быть выполнен без выполнения включенного.

Название прецедента определяется выполняемым действием и имеет форму глагола.

Обозначения диаграммы прецедентов



актер



прецедент

Пример диаграммы прецедентов



Лабораторная работа № 6. Представление варианта использования для проекта

Цель работы: познакомиться с понятием вариант использования, научиться формировать текстовое описание варианта использования и его графическое представление в виде диаграммы деятельности UML.

Задание

1. Изучить структуру шаблона текстового описания прецедента (Приложение 1).
2. На диаграмме прецедентов из лабораторной работы 5 выбрать любой прецедент.
3. По шаблону составить текстовое описание выбранного прецедента.
4. Построить диаграмму деятельности на основе описаний основного и альтернативных направлений (сценариев) варианта использования, исключений.

Справочный материал

Диаграмма деятельности — еще один тип диаграмм языка UML. Предназначена для графического описания последовательности действий (алгоритмов). Аналог — блок-схемы.

Диаграмма деятельности позволяет показывать линейные, разветвляющиеся алгоритмы, параллельное выполнение действий, сопоставлять действия и выполняющих их пользователей. Диаграмма может иметь следующие типы узлов: управляющие (решение), объектные (объекты), исполняемые (действия).

Обозначения диаграммы прецедентов

Управляющие узлы:



решение



параллельное исполнение



начало сценария

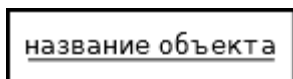


окончание сценария



остановка исполнения

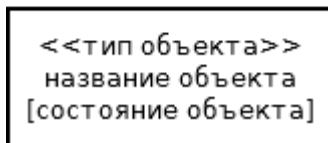
Объектные узлы:



объект

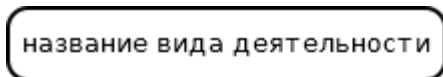


сигнал

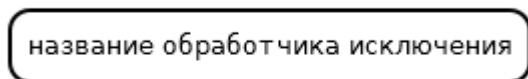


объект и его описание

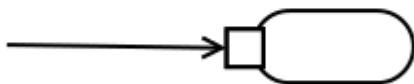
Исполняемые узлы



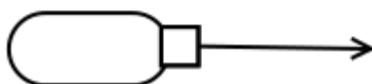
вид деятельности



обработка исключительной ситуации



узел, принимающий данные

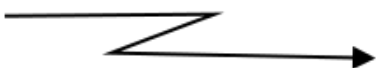


узел, передающий данные

Дуги



дуга между видами деятельности



дуга из области прерывания



именованная дуга

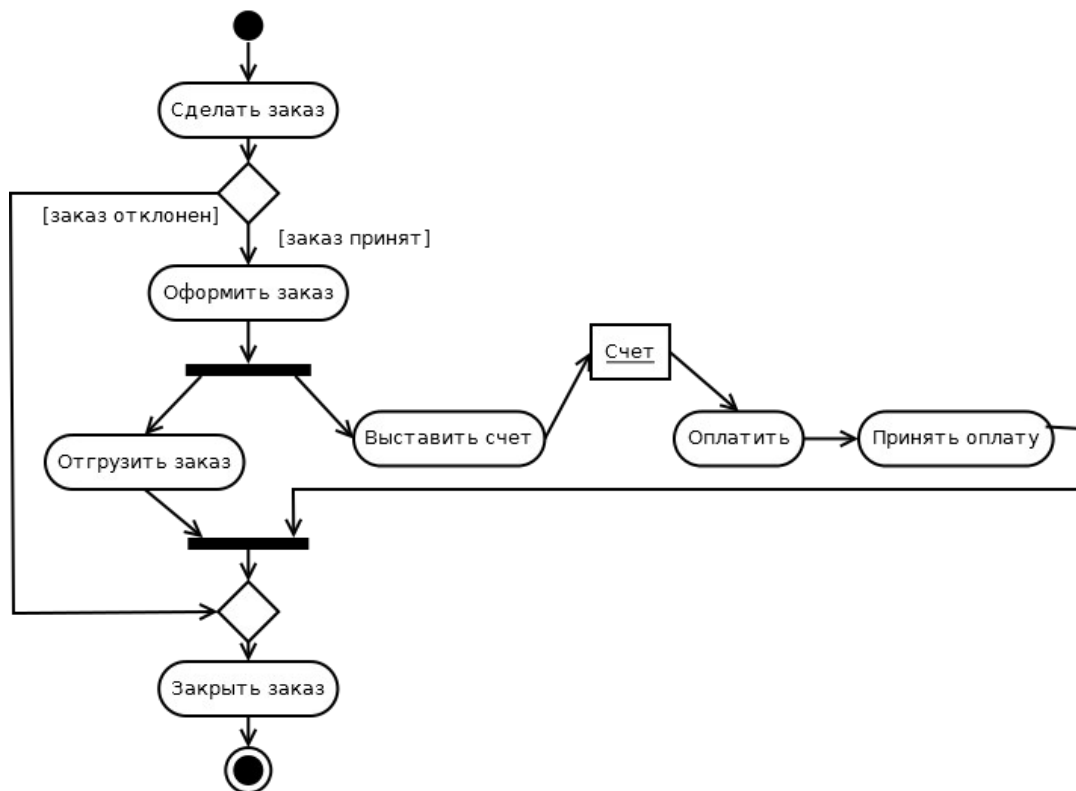
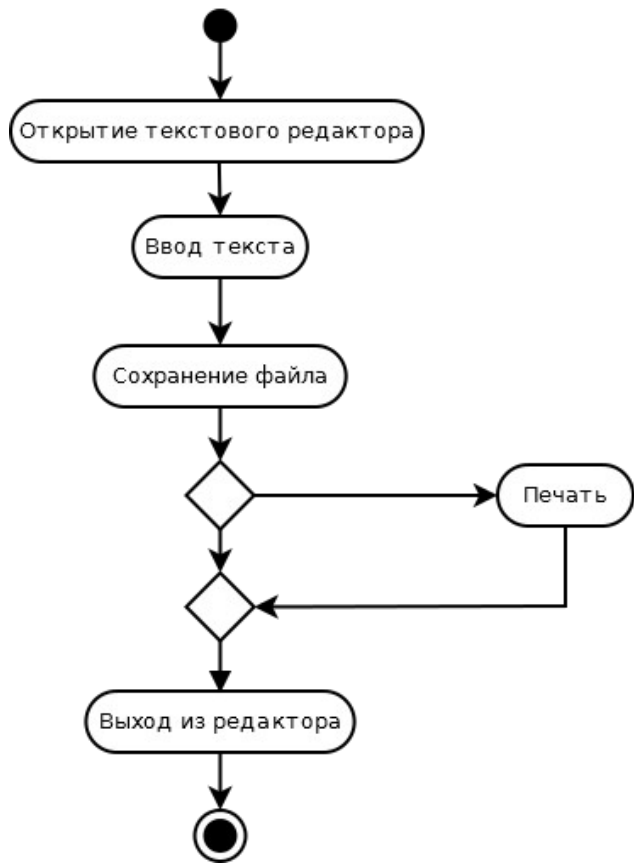


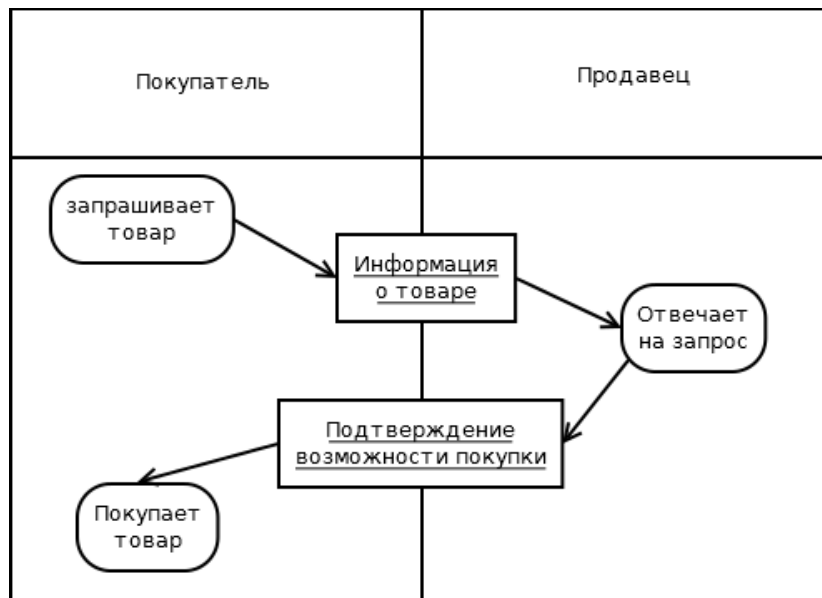
соединение дуг



примеры дуг

Примеры диаграммы прецедентов





Роли и объекты на диаграмме деятельности

Приложение 1. Шаблон варианта использования

- Идентификатор
- Имя (глагол+объект)
- Источник (автор)
- Дата создания
- Основное действующее лицо
- Дополнительно действующее лицо
- Триггер (действие, инициирующее вариант использования)
- Предварительные условия (начальное состояние)
- Выходные условия
- Нормальное направление варианта использования
- Альтернативное направление варианта использования
- Исключения
- Приоритет
- Частота использования
- Родительское бизнес-требование
- Бизнес-правила
- Специальные требования (Другая информация)
- Предположения