

ЛАБОРАТОРНАЯ РАБОТА №2

Разработка требований

Цель работы: выявление и описание пользовательских требований на основе составления вариантов использования (Use Cases).

Теоретические сведения

Требование (Requirement) – описание того, какие функции и с соблюдением каких условий должен выполнять программный продукт в процессе решения полезной для пользователя задачи.

Значение требований:

- Позволяют понять, что и с соблюдением каких условий система должна делать.
- Предоставляют возможность оценить масштаб изменений и управлять изменениями.
- Являются основой для формирования плана проекта (в том числе плана тестирования).
- Помогают предотвращать или разрешать конфликтные ситуации.
- Упрощают расстановку приоритетов в наборе задач.
- Позволяют объективно оценить степень прогресса в разработке проекта.

Работа над требованиями включает следующие этапы: выявление требований, анализ требований (моделирование бизнес-процессов, прототипирование интерфейсов, приоритезация требований, результат этапа – визуализация требований), документирование требований (результат этапа – спецификация), тестирование (валидация) требований. Работу с требованиями выполняет бизнес-аналитик.

Пользовательские требования (user requirements) описывают задачи, которые пользователь может выполнять с помощью разрабатываемой системы. Требования этого уровня могут быть использованы для оценки объёма работ, стоимости проекта, времени разработки. Пользовательские требования оформляются в виде вариантов использования (use cases), пользовательских историй (user stories), пользовательских сценариев (user scenarios).

Выявление и описание требований: Use Case и User Story.

Вариант использования (use case) продукта описывает последовательность взаимодействия системы и внешнего действующего лица. **Действующим лицом** (actor) может быть человек, другая система ПО или аппаратное устройство, взаимодействующее с системой для достижения некой цели.

Варианты использования меняют традиционный подход к сбору информации: пользователей не спрашивают, что с их точки зрения, должна

делать **система**, а выясняют, какие задачи собирается с ее помощью решать **пользователь**. Цель такого подхода — описать все подобные задачи. До включения каждого варианта использования в утвержденную версию требований заинтересованные в проекте лица проверяют, не выходит ли он за границы проекта. Теоретически в конечный набор вариантов использования должна входить вся желаемая функциональность системы. На практике добиться стопроцентного результата не представляется возможным, однако варианты использования помогут выполнить эту задачу полнее, чем какой-либо другой прием сбора информации.

Описание варианта использования включает следующие категории:

- уникальный идентификатор;
- имя, кратко описывающее задачи пользователя в формате «глагол + объект», например «разместить заказ»;
- краткое текстовое описание на естественном языке;
- список предварительных условий, которые должны быть удовлетворены до начала разработки варианта использования;
- выходные условия, описывающие состояние системы после успешного завершения разработки варианта использования;
- пронумерованный список действий, иллюстрирующий последовательность этапов взаимодействия лица и системы от предварительных условий до выходных условий.

Существует несколько сценариев варианта использования.

Один сценарий считается **нормальным направлением развития** (normal course) варианта использования, его также называют основным направлением, базовым направлением, нормальным потоком, основным сценарием, главным успешным сценарием и благоприятным путем. Нормальное направление для варианта использования «Запрос химиката» — запрос химиката, который есть на складе.

Другие допустимые сценарии из варианта использования, называются **альтернативными направлениями** (alternative courses) или **вторичными сценариями** (secondary scenarios). Они также могут привести к успешному выполнению задания и удовлетворяют выходным условиям варианта использования. Однако они представляют вариации решения задачи или диалоговой последовательности, необходимой для выполнения задачи. В определенной точке принятия решений в диалоговой последовательности нормальное направление может перейти в альтернативное, а затем вернуться обратно в нормальное.

Условия, препятствующие успешному завершению задания, называются **исключениями** (exceptions). Если в процессе сбора информации не указано, как обрабатывать исключение, то возможны два пути:

1. разработчики предложат лучший по их мнению способ обработки исключений;
2. при генерации пользователем неверного условия произойдет сбой

системы, так как никто не предусмотрел такой ситуации.

Иногда исключения рассматриваются как тип альтернативного направления, однако эти понятия следует разделять. Не обязательно реализовывать каждое альтернативное направление, которое определяют для варианта использования; кроме того, можно отложить его реализацию до следующего выпуска. Однако **необходимо** реализовать исключения, из-за которых завершение сценариев может оказаться неуспешным.

Расширение (extend) и включение (include).

При составлении вариантов использования часто можно столкнуться с ситуацией, когда альтернативное направление варианта использования само по себе можно выделить в автономный вариант использования. В таком случае можно расширить (extention) нормальное направление, включив этот отдельный вариант использования в нормальный поток.

Пример: Вариант использования "Запросить химикат" может включать в себя поиск по каталогу поставщика. Но при этом запросить химикат можно и без поиска по каталогам, а поиск по каталогу может выполняться как отдельная бизнес-задача пользователей. Поэтому логично расширить вариант использования "Запросить химикат" отдельным вариантом использования "Поиск по каталогам поставщика". Такая связь будет означать, что при выполнении варианта использования "Запросить химикат" может выполняться, а может и не выполняться вариант использования "Поиск по каталогам поставщика".

Иногда же несколько вариантов использования имеют общие наборы этапов. Чтобы избежать повторение этих этапов в каждом варианте использования, можно определить отдельный вариант использования и указать, что он включен (include) в другие варианты использования как подвариант.

Пример: Если покупатель совершает покупку товара, то он обязательно должен его оплатить. Нет случая, когда покупатель просто за что-то платит (т.е. оплата товара не является отдельной независимой деятельностью покупателя), но при этом процесс оплаты сам по себе достаточно сложный, включающий различные шаги и альтернативные варианты (оплата различными способами). Поэтому логично его выделить в отдельный вариант использования, при этом включить в вариант использования "Покупка товара". Такая связь будет означать, что при выполнении варианта использования "Купить товар" обязательно будет задействован и вариант использования "Оплатить товар".

Определение вариантов использования

Определить варианты использования можно несколькими способами:

- сначала определить действующие лица, а затем бизнес-процессы, в которых каждое лицо участвует;
- выразить бизнес-процессы в терминах определенных сценариев, обобщить сценарии в варианты использования и определить действующие

лица для каждого варианта; определить внешние события, на которые система должна реагировать, а затем соотнести эти события с участвующими лицами и определенными вариантами использования;

– определить вероятные варианты использования на основе функциональных требований, если какие-либо требования невозможно проследить до какого-либо варианта использования, необходимо задуматься нужны ли они.

Как правило, пользователи сначала определяют самые важные варианты использования, поэтому порядок предлагаемых тем позволит получить представление о приоритетах. Другой способ расстановки приоритетов — кратко описать каждый потенциальный вариант использования в том виде, как он был предложен.

Преимущества способа с применением вариантов использования

Преимущества применения вариантов использования в том, что каждый вариант сосредоточен на поставленной задаче и пользователе. Тщательное изучение этапов взаимодействия лица и системы помогает еще на ранних стадиях разработки выявить неясности и неточности, а также позволяет составить варианты тестирования на основе вариантов использования.

Способ с применением вариантов использования позволяет выявить функциональные требования, с помощью которых пользователи будут выполнять конкретные задачи.

Способ с применением вариантов использования облегчает расстановку приоритетов требований. Высшим приоритетом обладают те функциональные требования, которые созданы на основе вариантов использования с высшим приоритетом. Высший приоритет назначается по следующим причинам:

- варианты использования описывают один из основных бизнес-процессов, активизируемых системой;
- многие пользователи часто обращаются к ним;
- их запросил привилегированный класс пользователей;
- они предоставляют возможности, необходимые для соответствия требованиям;
- функции других систем зависят от их наличия.

Существуют также и преимущества технического характера. С помощью варианта использования можно выявить некоторые важные объекты предметной области и их взаимоотношения. Разработчики, использующие объектно-ориентированные методы проектирования, могут преобразовать варианты использования в объектные модели, такие, как диаграммы классов и диаграммы последовательностей.

Пример выполнения лабораторной работы

Необходимо составить варианты использования функциональности «Сообщество» социальной сети.

Описание одного из вариантов использования функциональности «Сообщество» социальной сети приведено в таблице 1.

Таблица 1 – Описание варианта использования «Вступить в сообщество»

ID	2 UC
Наименование	Вступить в сообщество
Учасники	Зарегистрированные пользователи ВК
Предусловие	Пользователь должен быть зарегистрирован в социальной сети ВК
Триггер	Пользователь нажимает категорию меню "Мои сообщества"
Основной сценарий	<p>1. Пользователь осуществляет поиск сообщества (1 UC).</p> <p>2. Пользователь осуществляет подписку на сообщество кликнув кнопку "Подписаться" (см. рис. Р6)</p> <p>3. Система оповещает пользователя о вступлении в сообщество сообщением "Вы подписаны", (см. рис. Р7)</p>
Альтернативный сценарий	<p>2А. Пользователь переходит на страницу сообщества.</p> <p>2Б. Пользователь кликает на кнопку "Подписаться" либо "Вступить в группу".</p> <p>3А. Система оповещает пользователя о вступлении в сообщество одним из следующих сообщений "Вы подписаны на новости", "Вы состоите в группе".</p>
Исключения	<p>3А_И1. Сообщество является закрытым</p> <p>3А_И1-1. Пользователь переходит на страницу сообщества.</p> <p>3А_И1-2. Пользователь подает запрос на вступление в сообщество (кликает на кнопку "Подать заявку")</p> <p>3А_И1-2а. Система отправляет заявку пользователя администратору сообщества.</p> <p>3А_И1-2б. Администратор рассматривает поданную заявку и принимает ее либо отклоняет.</p>

Порядок выполнения работы

1. Получить задание у преподавателя.