



Изучаемый курс
**«ТЕХНОЛОГИЯ РАЗРАБОТКИ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»**

Преподаватель дисциплины:
Нефедова Людмила Петровна

Специальность:
**09.02.07
информационные системы и
программирование**

Учебный материал

- «Осуществление интеграции программных модулей» Г.Н. Федорова учебник для студентов учреждений СПО - М.: «Академия», 2019.-288с. **Стр 36-60.**
- 1. «Технология разработки программных продуктов» А.В. Рудаков
- 2. «Технологии разработки программного обеспечения» С.А. Орлова
- Ресурс НПК на портале (платформа **MOODLE**). «Занятие 5.doc»
- Прочие Интернет-ресурсы

Занятие 5.

Управление требованиями.

- Понятия требований, классификация, уровни требований.
- Методологии и стандарты, регламентирующие работу с требованиями

Часть 1

**Понятия требований,
классификация, уровни
требований.**

Определения



1. ***Спецификация** – точное формализованное описание функций и ограничений разрабатываемого ПП.
2. **Требование** – условие или характеристика, которой должно удовлетворять программное обеспечение или свойства, которым оно должно удовлетворять.
3. **Спецификация требований** – основной документ, играет определяющую роль по отношению к другим элементам плана разработки.

Первичные требования заказчика

- Разработка любого ПО начинается с анализа требований к будущему ПП. В результате анализа получают спецификации разрабатываемого ПО.
- Изначально собираемые требования называют **первичными требованиями (ПТЗ)**. ПТЗ – это протоколы совещаний, интервью с заказчиками и пользователями, оригиналы документов, аналоги ПП. В конечном итоге ПТЗ не должны противоречить друг другу и зафиксированы все.

Управление требованиями

На практике процесс сбора, анализа и обработки растянут во времени на протяжении всего жизненного цикла ПО. Необходим учет всех требований, контроль, обработка. Такая работа называется **управлением требованиями**.

У требований устанавливаются приоритеты: «необходимо выполнить», «следует выполнить», «можно выполнить».

Цель Управления требованиями

- соглашения между заказчиком, пользователем и исполнителем
- лучшего понимания задач разработчиком ПО
- установления границ ПО (требования к аппаратуре компьютера, операционной среде, возможностям ПО)
- планирование

Виды требований

- **Функциональные требования** регламентируют функционирование или поведение системы (behavioral requirements). Функциональные требования отвечают на вопрос «**что должна делать система**» в тех или иных ситуациях. Функциональные требования определяют основной «фронт работ» Разработчика.
- Функциональные требования записываются, как правило, при помощи **предписывающих правил**: «система должна позволять кладовщику формировать приходные и расходные накладные».
- Другим способом являются так называемые варианты использования (uses cases) – популярный и весьма продуктивный способ представления требований.

• Это основной представляющий вид требований

- **Нефункциональные требования**, соответственно, регламентируют внутренние и внешние условия или атрибуты функционирования системы.

Основные группы нефункциональных требований:

- Внешние интерфейсы (External Interfaces),
- Атрибуты качества (Quality Attributes),
- Ограничения (Constraints).

Среди *внешних интерфейсов* в большинстве современных АИС наиболее важным является интерфейс пользователя (User Interface, UI). Кроме того, выделяются интерфейсы с внешними устройствами (аппаратные интерфейсы), программные интерфейсы и интерфейсы передачи информации (коммуникационные интерфейсы).

Основные *атрибуты качества*: Применимость, Надежность, Производительность, Эксплуатационная пригодность,

Ограничения - формулировки условий, модифицирующих требования или наборы требований, сужая выбор возможных решений по их реализации, выбор платформы реализации и/или развертывания (протоколы, серверы приложений, баз данных, ...), которые в свою очередь могут относиться например к внешним

Уровни требований


Верхний уровень.

Бизнес – требования (business requirements) – общее представления о функциональности системы.



Средний уровень.

Уровень требований пользователей (user requirements).



Низкий уровень.

Функциональный (functional requirements) – детализация пользовательских запросов.

Бизнес-требования

высказывают те, кто финансируют проект, покупатели системы, топ-менеджеры, отдел маркетинга.

Пример бизнес-требования

на разработку сайта поиска авиабилетов:

«Посетителю сайта ресурс обеспечивает возможность подобрать оптимальный по цене маршрут, чтобы сэкономить деньги. Также позволяет сэкономить время планирования поездки, потому что поиск авиабилетов на разных сайтах разных авиакомпаний и их сравнение всегда требует большого

Уровень требований пользователей

Например.

Можно описать пошаговый сценарий покупки авиабилета на выбранный маршрут, и сам этот сценарий, собственно, и будет формой представления пользовательских требований.

Требования пользователей часто бывают *плохо структурированными, дублирующимися, противоречивыми.*

Поэтому для создания системы важен третий уровень, в котором осуществляется *формализация требований.*

Функциональный

Данные требования определяют функциональность ПО, которую разработчики должны построить, чтобы пользователи смогли выполнить свои задачи в рамках бизнес-требований.

Пример:

«Система должна по электронной почте отправлять пользователю подтверждение о заказе».

Пример различий между пользовательскими и функциональными требованиями:

Пользовательское: «Система должна выводить отчеты на печать»

Функциональное: «Система должна обеспечивать вывод отчетов на печать, обеспечивать возможность выбора и настройки локального или сетевого принтера, выбора ориентации бумаги».

Характеристика	Объяснение
Единичность	Требование относится только к одному свойству
Завершённость	Требование полностью определено в одном месте и вся необходимая информация присутствует.
Последовательность	Требование не противоречит другим требованиям и полностью соответствует внешней документации.
Атомарность	Требование «атомарно». То есть оно не может быть разбито на ряд более детальных требований без потери завершённости.
Отслеживаемость	Требование полностью или частично соответствует деловым нуждам как заявлено заинтересованными лицами и документировано.
Актуальность	Требование не стало устаревшим с течением времени.
Выполнимость	Требование может быть реализовано в пределах проекта.
Понятность	Требование должно пониматься однозначно всеми <i>Требование кратко определено без обращения к техническому жаргону, акронимам и другим скрытым формулировкам. Оно выражает объективные факты, не субъективные мнения. Возможна одна и только одна интерпретация. Определение не содержит нечётких фраз. Использование отрицательных утверждений и составных утверждений запрещено.</i>
Обязательность	Требование представляет определённую заинтересованным лицом характеристику, отсутствие которой приведёт к неполноценности решения, которая не может быть проигнорирована. <i>Необязательное требование — противоречие самому понятию требования.</i>
Проверяемость	Реализованность требования может быть определена через один из четырёх возможных методов: осмотр, демонстрация, тест или анализ.
Полнота	Что система должна делать

Источники требований

- Федеральное и муниципальное отраслевое законодательство (конституция, законы, распоряжения)
- Нормативное обеспечение организации (регламенты, положения, уставы, приказы)
- Текущая организация деятельности объекта автоматизации
- Модели деятельности (диаграммы бизнес-процессов)
- Представления и ожидания потребителей и пользователей системы
- Журналы использования существующих программно-аппаратных систем
- Конкурирующие программные продукты

Этапы сбора и анализа требований к ПП

№	Название этапа	Действующие лица:	Цель:	Результат:
1.	Определение концепции продукта	Аналитик – Представитель заказчика (инвестор)	формирование бизнес – требований, выработка единого видения проекта;	Принятие решения о разработке проекта
2.	Сбор требований	Аналитик – Представитель заказчика (эксперт, пользователь)	Определение функций ПП и способов их интеграции в существующие процессы	Список функций и их приоритетов
3.	Анализ требований	Аналитик	Структуризация собранных требований	Окончательный список структурированных, не дублируемых функций
4.	Проектирование системы	Аналитик, проектировщик	Создание проектного решения о функциональности ПП, моделирование требований	Техническое задание – документ, описывающий поведение

Анализ и структурирование первичных требований

- **Анализ первичных требований** – обобщенное описание ПО, описания отдельных функций нет или не детализировано. Проводится совместно с заказчиком, выяснение возможности создания данного ПО, определение сроков, способов.
- **Структурирование требований:**
 - Общее назначение, особенности аппаратных средств, выделение компонент
 - Детализация обобщенной структуры: уточнение описаний взаимосвязей компонент, детальное описание каждого, общее описание каждой функции, анализ требований с точки зрения возможности и сроков.
 - Составление описания отдельной функции каждого элемента: входные и выходные данные, действия над данными, способы тестирования. Глубина детализации определяется руководителем проекта, разработчик и заказчик должны получить ясное представление.

Моделирование предметной области

- Модель – некоторая система, имитирующая структуру и функционирование исследуемой предметной области, основным требованием – адекватность этой области.
- Модель должна отражать все аспекты ПО , необходима на всех этапах ЖЦ.
- Без моделирования велика вероятность допущения большого числа ошибок, приводящих к экономическим потерям.

Требования к модели

- Формализация
- Понятность для заказчика
- Реализуемость (наличие средств реализации)
- Возможность оценки эффективности реализации модели предметной области на основании методов и рассчитываемых показателей.

Необходимо построение:

- **Объектной модели** - взаимодействие материальных и информационных объектов.
- **Функциональной модели** – взаимосвязь функций (действий) по преобразованию объектов в процессах
- **Технической модели** – топология расположения, способы коммуникации комплекса технических средств.
- Модель управления (возможно) – воздействие бизнес-правил, событий на выполнение процессов.
- **Организационная** - воздействие организационных единиц.

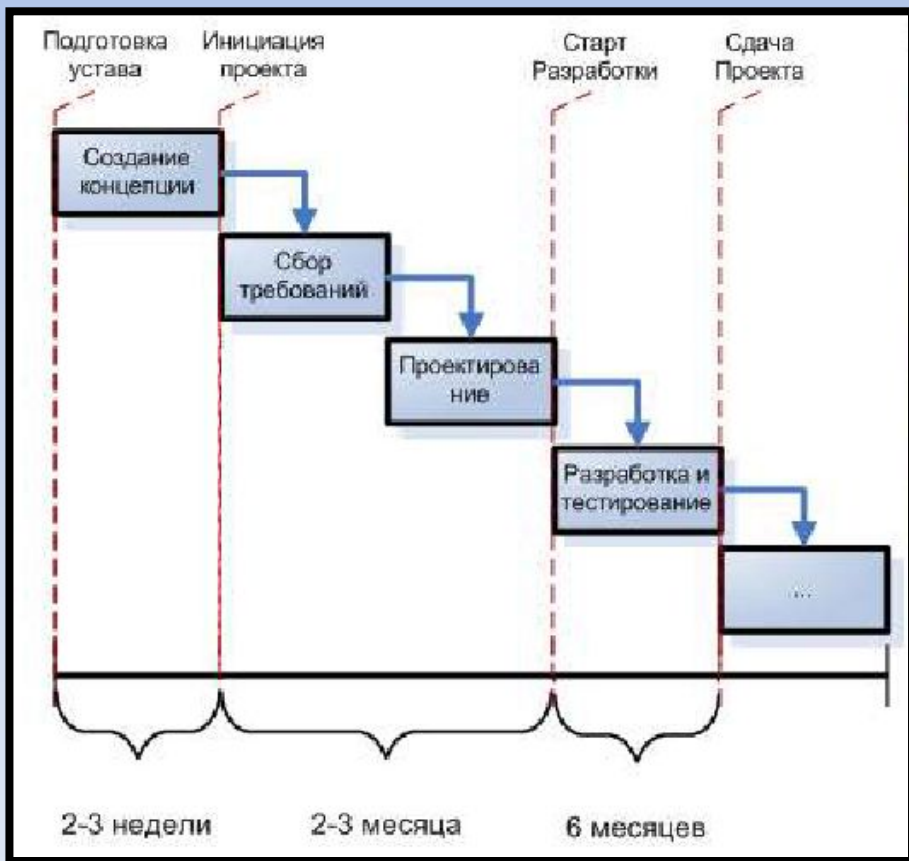
Критерий адекватности модели – функциональная полнота.

Моделирование

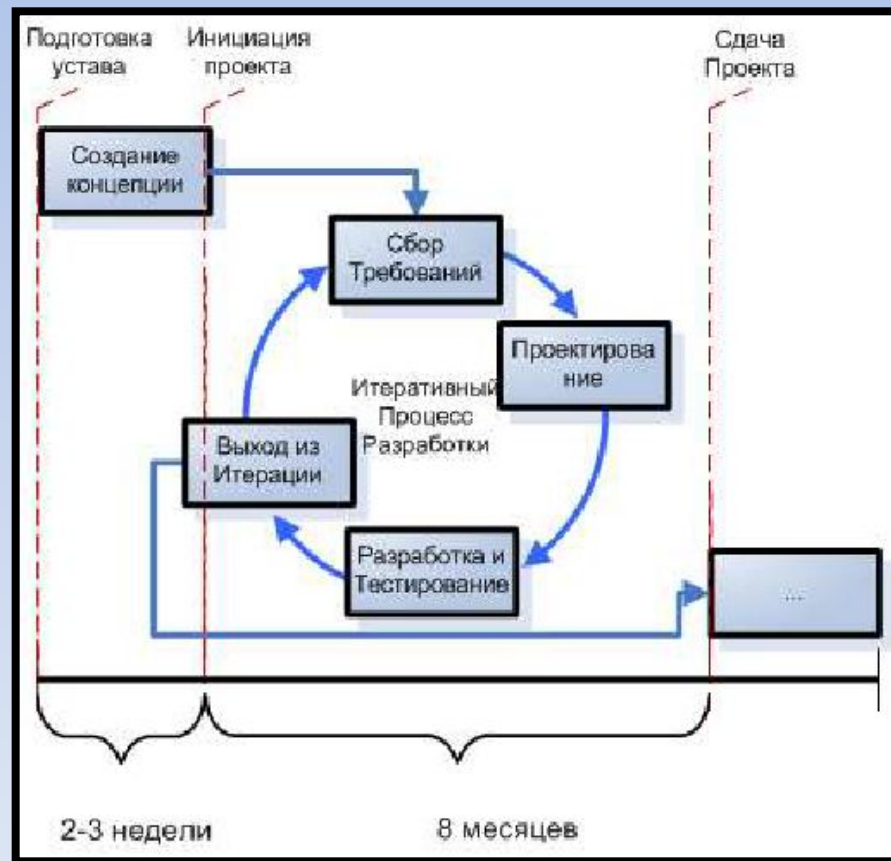
- Для описания модели используется **Язык моделирования** – графическая нотация для описания проекта или синтаксический язык моделирования. Это средство формализации, понятен пользователю и средство проектирования решений.
- Нотация – совокупность графических объектов, используемых в модели.
- Уровни моделей:
 - **Верхний уровень** – определение требований, состав компонентов: объектов, функций, событий
 - **Концептуальный уровень** – спецификация требований, характер взаимодействий
 - **Внутренний уровень** – реализация требований, определение программно-технических средств.
- Сложная система разбивается (**декомпозиция**) на небольшие подсистемы, каждая из которых рассматривается независимо друг от друга

Интеграция процессов формирования требований в ЖЦ

Линейный подход (каскадная модель)



Итерационный подход (RAD, Agile - модели)



Часть 2

**Методологии и стандарты,
регламентирующие работу с
требованиями**

Методологии и стандарты, регламентирующие работу с требованиями

- **Разработки IEEE:**

- IEEE 1362 “Concept of Operations Document”. Kurt Bittner, Ian Spence. Use Case Modeling, 2002.
- IEEE 1233 «Guide for Developing System Requirements Specifications».
- IEEE Standard 830-1998, «IEEE Recommended Practice for Software Requirements Specifications»
- IEEE Standard Glossary of Software Engineering Terminology/IEEE Std 610.12- 1990
- IEEE Guide to the Software Engineering Body of Knowledge (1) - SWEBOOK®, 2004.

- **Отечественные ГОСТ:**

- ГОСТ 34.601-90. Информационная технология. Автоматизированные системы. Стадии создания.
- ГОСТ 34.602-89. Информационная технология. Техническое задание на создание автоматизированной системы
- ГОСТ 19.201-78. Единая система программной документации. Техническое задание. Требования к содержанию и оформлению.

Методы проведения обследований

- **По цели.**
 - Метод локального проведения обследования для отдельной задачи или комплекса задач.
 - Метод системного обследования объекта для изучения всего объекта
- **По числу исполнителей, проводящих обследование**
 - Индивидуальное обследование
 - Бригадное: бригады-исполнители и бригада-координатор
- **По степени охвата предметной области.**
 - Метод сплошного обследования: все подразделения производства
 - Выборочное: при наличии типовых объектов
- **Методы последовательного и параллельного проведения работ**
 - Метод последовательного проведения работ: сначала собирают данные, потом изучают (часто применяют при отсутствии опыта)
 - Методы параллельного проведения работ: одновременно собирают данные и изучают (сокращает время перед проектной стадией, повышает качество результатов)

Выполнение работ по обследованию предметной области и сбору материалов можно выполнять на основе **предварительного выбора методов** из групп:

- Методы сбора силами проектировщиков (беседы опросы, анализ, хронометраж, фотографии)
- Методы сбора силами специалистов по предметной области (ведение тетради-дневника на осуществляемые работы, необходимо выявлять состав операций и получаемые при этом документы)
- Методы бесед и консультаций с руководителями (обычная беседа с руководителями разных уровней и подразделений, группами пользователей, специалистами по этим проблемам)
- Метод опроса исполнителей на рабочих местах (по списку сотрудников, готовится перечень вопросов о назначении работ, порядке выполнения)
- Метод анализа операций (расчленение делового процесса на составные части, задачи, расчеты, операции. Анализируется каждая часть, выявляется повторяемость операций, многократное обращение к одной и той же, их зависимость)

Документирование требований

Документирование требований логически завершает процесс формирования требований, подлежат обязательному двустороннему утверждению - Заказчиком и Разработчиком. После этого они вступают в свою полную силу.

Спецификация (документирование) требований к ПП (методология RUP)

К основным документам, описывающим требования к ПП, относят:

❖ Концепция

Определяет бизнес – требования и глобальные цели проекта;

❖ Словарь предметной области (глоссарий)

Определяет общую терминологию для всех моделей и описаний требований;

❖ Варианты использования

Документирует и моделирует функциональные требования к ПП;

❖ Последовательные спецификации (технические

Составление спецификаций

1. Анализ требований => 2. Спецификации
разрабатываемого ПО: выполняют декомпозицию и содержательную постановку решаемых задач, уточняют их взаимодействие, определяют эксплуатационные ограничения.

- **Спецификация** – точное формализованное описание функций и ограничений разрабатываемого ПП.
- **Функциональная спецификация** в разработке ПО – документ, описывающий требуемые характеристики системы (функциональность)
- **Эксплуатационная спецификация** – описание правил использования ПО, определяют требования к техническим средствам, надежности, безопасности...

2. Определения спецификаций =>

3. Общая модель предметной области

- В целом в процессе определения спецификаций строится **общая модель предметной области**, как части реального мира, с которой будет взаимодействовать разрабатываемое ПО, конкретизируют его основные функции.
- **Функциональные спецификации** должны быть полными (вся важная информация, нет упущений) и точными (однозначны для заказчика и разработчика).
- **Функциональная спецификация состоит из трех частей**
 - **Описание внешней информационной среды** (каналы ввода-вывода, информационные объекты, использующие это ПО)
 - **Определение функций ПО** (множество состояний, обозначение функций, специфицируются входные данные и результаты)
 - **Определение исключительных ситуаций** (ситуация и

- Функциональные спецификации = **Перечень функций + состав обрабатываемых данных**
- Различаются Функциональные спецификации установленным **приоритетом требований**
- Все изменения параметров отдельных требований или пожелания со стороны заказчика оформляются в **приложении** к спецификациям требований.
- **Точные требования** можно определить, разработав некоторую формальную модель разрабатываемого ПО

Конструирование прототипа

- одна из самых важных стадий; быстрая «черновая» реализация базовой функциональности
- **Прототип интерфейса** пользователя (UI), оперативно созданный схематично, дает возможность представления части системы (подтвердит - да или нет). Имитируются расчеты, запросы во внешние системы, реализуется часть кода, отвечающая за перемещение между экранами – дает понимание реакции системы на действия пользователя.
- **Возможность выбрать** из предложенного (формулировки требований, реализации пользовательского интерфейса)
- **Определение риска невозможности реализации** (функциональные +не функциональные требования, например быстроедействие с учетом ограничений среды реализации)

Виды прототипов

- *Горизонтальный* моделирует интерфейс, не затрагивая логику обработки и структур данных.
- ✓ Поясняет нечеткие требования.
- ✓ Не обязательно использовать программную среду, что и для конечной разработки
- ✓ Тексты отражают реальную специфику предметной области
- ✓ Реализуется часть кода, отвечающая за перемещение между экранами
- ✓ Используется **модель диаграммы состояний**, где разным экранам(окнам) сопоставляются **состояния**, а активным элементам управления, вызывающим открытие и закрытие интерфейсных элементов - **переходы**.

- **Вертикальный** моделирует не интерфейс, а вертикальный срез системы, все уровни.
- ✓ Обязательно использовать языки и ту же программную среду что, и для конечной разработки
- ✓ Для анализа применимости, проверки архитектурных кондиций.
- **Одноразовый** – исследовательский, быстрый, для RAD-технологии, код сырой
- **Эволюционный** - первое приближение системы, тщательная разработка, применяются технологические методы, тестирование результатов. Позволяет уточнить требования. Начальная версия продукта для демонстрации концепции. В результате возможно упрощение требований, упрощение функционала, анализ рисков на этапе планирования. Уменьшает стоимость. Существуют **риски** упущения требований, которые нужно устранять на более поздних стадиях процесса разработки.

Этапы процесса разработки спецификации*

Этапы разработки спецификации требований в случае, когда для ее проверки используется прототип системы.



Вопросы для самоконтроля

1. Назовите основные цели, преследуемые при анализе требований в проектах.
2. Перечислите Виды требований.
3. Перечислите Уровни требований пользователей
4. Назовите методы выявления требований.
5. Перечислите задачи, которые решаются на стадии анализа требований.
6. Перечислите Методы проведения обследований
7. При использовании методов по обследованию предметной области можно использовать Группы мероприятий. Перечислите эти Группы.
8. Что называется прототипированием?
9. Преимущества и недостатки использования прототипирования.

Дополнительные источники

- <https://intuit.ru/studies/courses/11876/1156/lecture/18252%2525253Fpage%2525253D2>
- <https://merehead.com/ru/blog/proof-concept-software-development/>