

Лекция 2: Жизненный цикл программного обеспечения, требования

Юрий Литвинов
y.litvinov@spbu.ru

27.02.2025

Виды деятельности при разработке ПО

- ▶ Возникновение и исследование идеи
- ▶ Анализ и сбор требований (пилотный проект)
- ▶ Планирование и проектирование
- ▶ Разработка
- ▶ Отладка и тестирование
- ▶ Сдача
- ▶ Сопровождение

Жизненный цикл ПО

- ▶ Период времени от возникновения идеи до прекращения использования
- ▶ Последовательность этапов
 - ▶ Состав и последовательность работ
 - ▶ Получаемые результаты
 - ▶ Методы и средства
 - ▶ Роли и ответственности
 - ▶ ...
- ▶ Модели жизненного цикла

Самая популярная модель разработки ПО

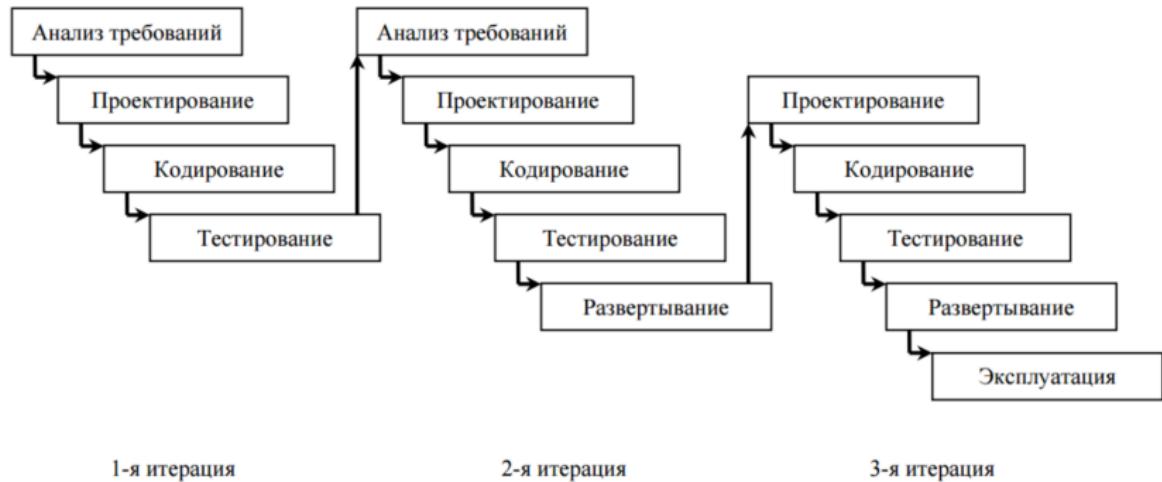


Водопадная (каскадная) модель

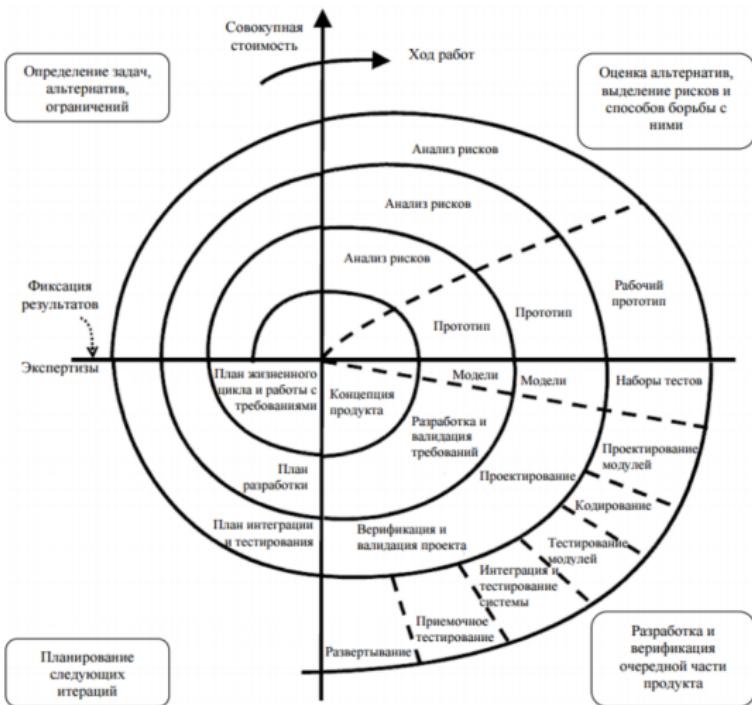


B. Ройс, 1970г, «Managing the Development of Large Software Systems»

Итеративная модель



Спиралевидная модель

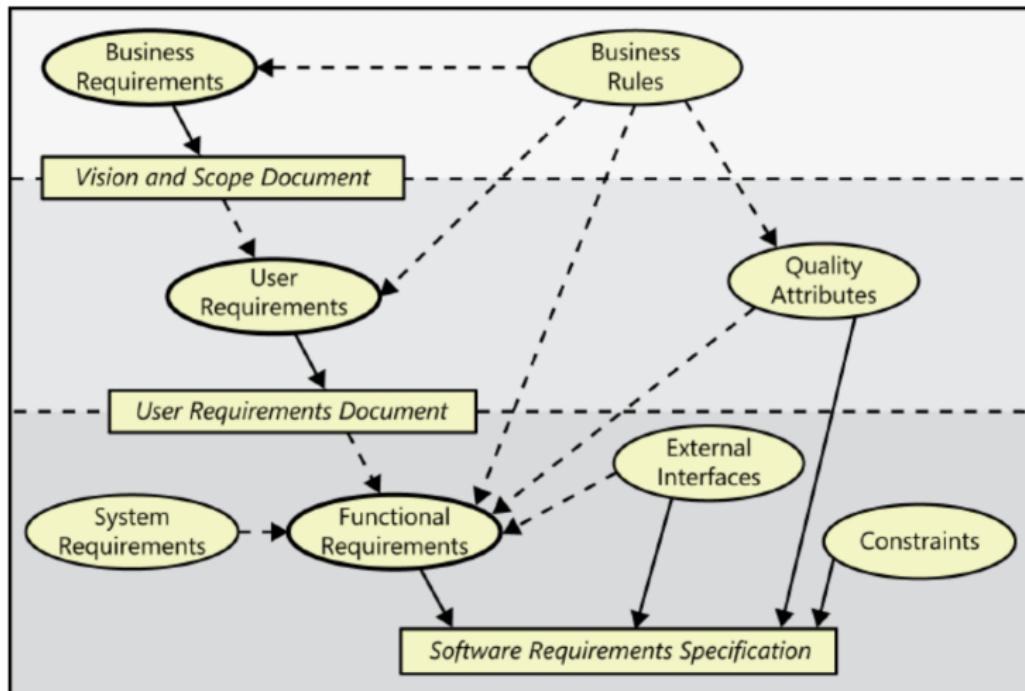


Требования

Требование — это любое условие, которому должна соответствовать разрабатываемая система или программное средство

- ▶ Возможности
- ▶ Ограничения

Виды требований



Пример

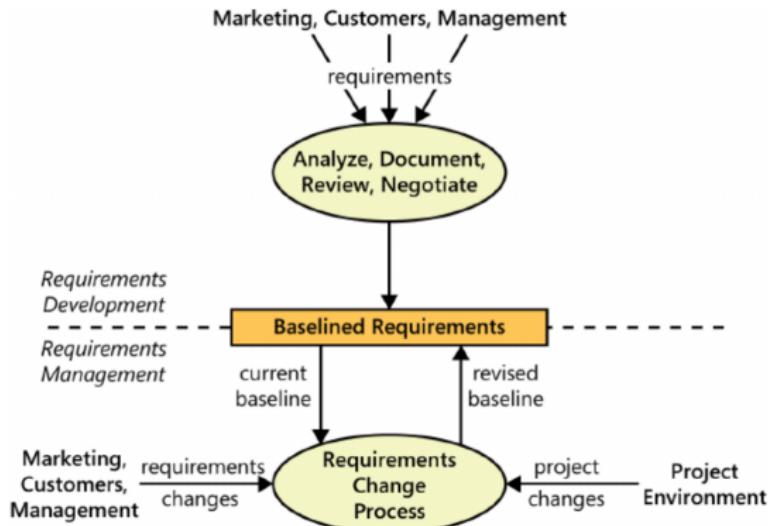
- ▶ Бизнес-требование:
 - ▶ продукт позволит пользователям эффективно исправлять орфографические ошибки в тексте
- ▶ Требования пользователей:
 - ▶ сценарии «Найти орфографическую ошибку» или «Добавить слово в общий словарь»
- ▶ Функциональные требования:
 - ▶ поиск и выделение слова с ошибкой
 - ▶ отображение диалогового окна с фрагментом текста с ошибочным словом
 - ▶ полнотекстовая замена слова с ошибкой
- ▶ Атрибуты качества:
 - ▶ простота использования («эффективность»)

Требования к требованиям

- ▶ Единичность
- ▶ Завершенность
- ▶ Непротиворечивость
- ▶ Атомарность
- ▶ Отслеживаемость
- ▶ Актуальность
- ▶ Выполнимость
- ▶ Недвусмысленность
- ▶ Обязательность
- ▶ Проверяемость

Работа с требованиями

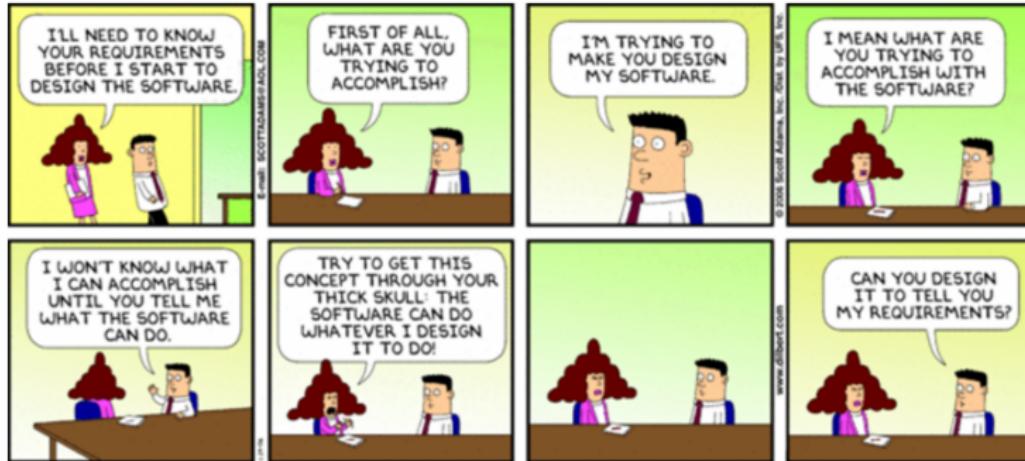
- ▶ разработка требований
 - ▶ выявление требований
 - ▶ анализ требований
 - ▶ спецификация требований
 - ▶ проверка требований
- ▶ управление требованиями



Выявление требований

- ▶ Определение процесса формулирования требований
- ▶ Определение классов пользователей и их характеристик
- ▶ Выбор типичного пользователя в каждом классе пользователей
- ▶ Организация фокус-групп/интервью типичных пользователей
- ▶ Наблюдение за пользователями на рабочих местах
- ▶ Изучение отчетов о проблемах работающих систем
- ▶ Определение системных событий и реакции на них
- ▶ Создание документа об образе и границах проекта

Выявление требований в реальности



Анализ требований

- ▶ Анализ осуществимости требований
- ▶ Определение приоритетов требований
- ▶ Моделирование требований
- ▶ Создание словаря терминов
- ▶ Создание контекстной диаграммы
- ▶ Создание пользовательского интерфейса и технических прототипов

Проверка требований

- ▶ Изучение документов с требованиями
- ▶ Определение критериев приемлемости
- ▶ Тестирование требований

Основные риски, связанные с требованиями

- ▶ Недостаточное вовлечение пользователей
- ▶ Игнорирование классов пользователей
- ▶ Разрастание требований пользователей
- ▶ Двусмысленность требований
- ▶ Излишняя дополнительная функциональность
- ▶ Урезанная спецификация
- ▶ Небрежное планирование

Трудовые функции аналитика

Профстандарт 06.022 «Системный аналитик»

- ▶ Разработка и сопровождение требований к отдельным функциям системы
- ▶ Разработка и сопровождение требований и технических заданий на разработку и модернизацию систем и подсистем малого и среднего масштаба и сложности
- ▶ Концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности
- ▶ Управление аналитическими работами и подразделением

Квалификационные уровни с 4 по 7

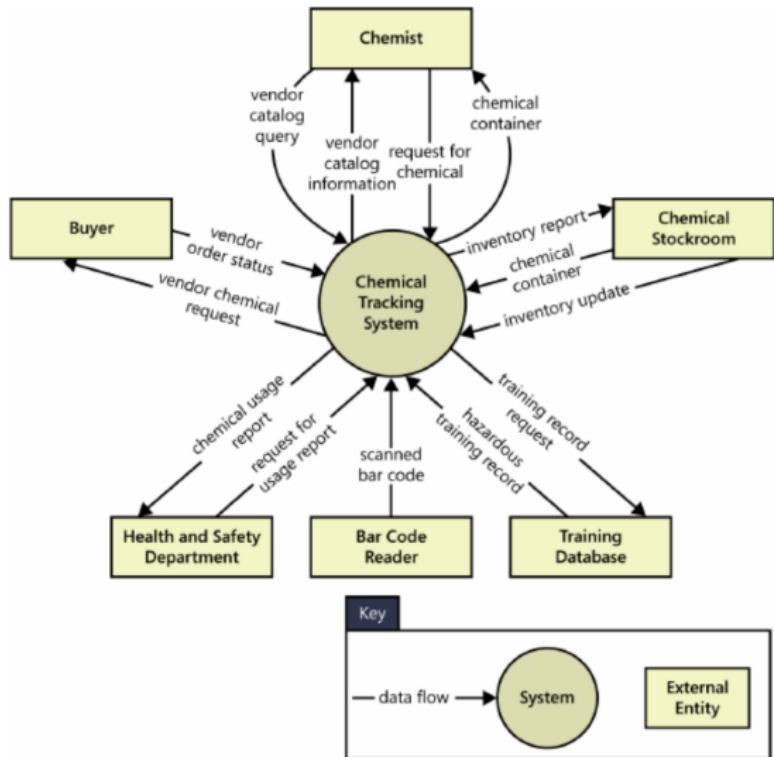
Создаваемые документы

- ▶ Документ об образе и границах системы
- ▶ Глоссарий
- ▶ Модель требований
- ▶ Прототип пользовательского интерфейса
- ▶ Спецификация требований

Документ об образе и границах проекта

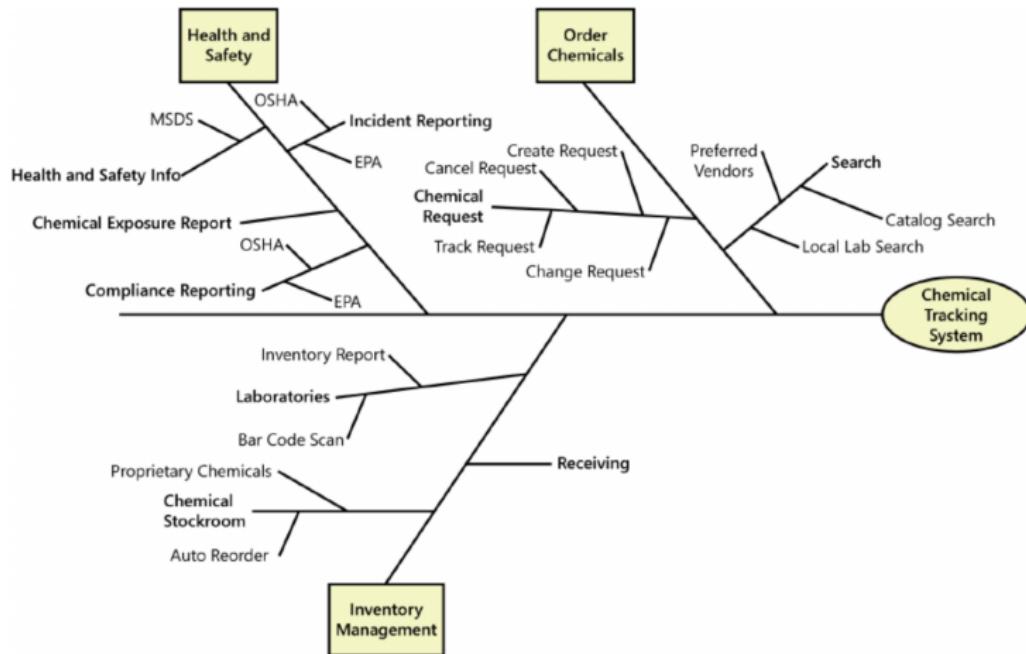
- ▶ Бизнес-требования
 - ▶ Контекст
 - ▶ Бизнес-возможности
 - ▶ Бизнес-цели и критерии успеха
 - ▶ Потребности клиента или рынка
 - ▶ Бизнес-риски
- ▶ Project vision
 - ▶ Положение об образе проекта
 - ▶ Основные функции
 - ▶ Предположения и зависимости
- ▶ Масштабы и ограничения проекта
 - ▶ Основная функциональность
 - ▶ Объем первоначальной версии
 - ▶ Объем последующих версий
 - ▶ Ограничения и исключения
- ▶ Бизнес-контекст
 - ▶ Профили заинтересованных лиц
 - ▶ Приоритеты проекта
 - ▶ Операционная среда

Контекстная диаграмма



Задание границ системы
и связи с внешним миром

Feature tree

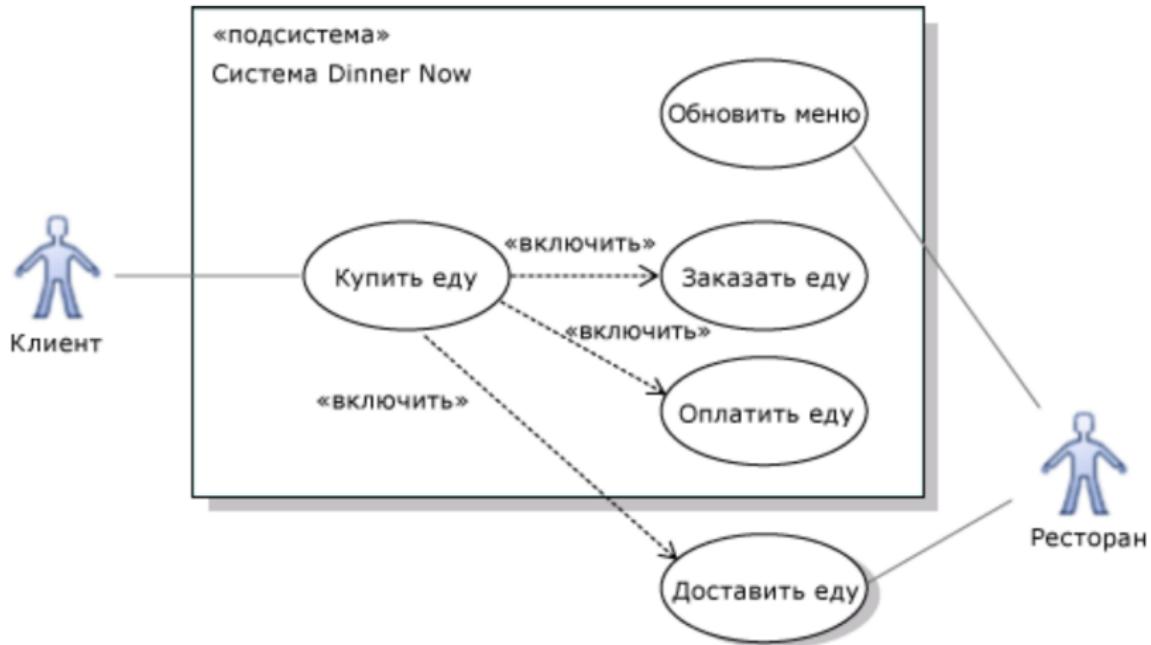


Моделирование требований

- ▶ Диаграммы случаев использования
 - ▶ Актёры (роли)
 - ▶ кто-то или что-то вне системы, влияющий или оказывающий влияние
 - ▶ люди или программные системы
 - ▶ Случаи (сценарии) использования
 - ▶ цель пользователя системы
 - ▶ действие-реакция для достижения цели
- ▶ Общее представление функциональных требований



Задание границ проекта



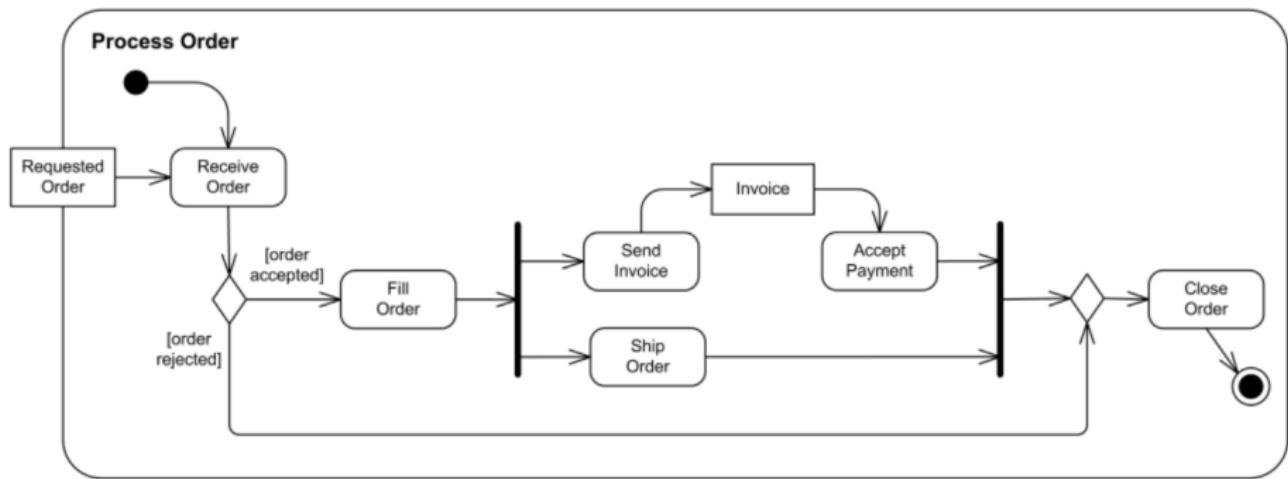
Сценарий использования (1)

ID and Name:	UC-4 Request a Chemical		
Created By:	Lori	Date Created:	8/22/13
Primary Actor:	Requester	Secondary Actors:	Buyer, Chemical Stockroom, Training Database
Description:	The Requester specifies the desired chemical to request by entering its name or chemical ID number or by importing its structure from a chemical drawing tool. The system either offers the Requester a container of the chemical from the chemical stockroom or lets the Requester order one from a vendor.		
Trigger:	Requester indicates that he wants to request a chemical.		
Preconditions:	PRE-1. User's identity has been authenticated. PRE-2. User is authorized to request chemicals. PRE-3. Chemical inventory database is online.		
Postconditions:	POST-1. Request is stored in the CTS. POST-2. Request was sent to the Chemical Stockroom or to a Buyer.		
Normal Flow:	4.0 Request a Chemical from the Chemical Stockroom 1. Requester specifies the desired chemical. 2. System lists containers of the desired chemical that are in the chemical stockroom, if any. 3. System gives Requester the option to View Container History for any container. 4. Requester selects a specific container or asks to place a vendor order (see 4.1). 5. Requester enters other information to complete the request. 6. System stores the request and notifies the Chemical Stockroom.		

Сценарий использования (2)

Alternative Flows:	4.1 Request a Chemical from a Vendor <ol style="list-style-type: none"> 1. Requester searches vendor catalogs for the chemical (see 4.1.E1). 2. System displays a list of vendors for the chemical with available container sizes, grades, and prices. 3. Requester selects a vendor, container size, grade, and number of containers. 4. Requester enters other information to complete the request. 5. System stores the request and notifies the Buyer.
Exceptions:	4.1.E1 Chemical Is Not Commercially Available <ol style="list-style-type: none"> 1. System displays message: No vendors for that chemical. 2. System asks Requester if he wants to request another chemical (3a) or to exit (4a). <ol style="list-style-type: none"> 3a. Requester asks to request another chemical. 3b. System starts normal flow over. 4a. Requester asks to exit. 4b. System terminates use case.
Priority:	High
Frequency of Use:	Approximately 5 times per week by each chemist, 200 times per week by chemical stockroom staff
Business Rules:	BR-28, BR-31
Other Information:	The system must be able to import a chemical structure in the standard encoded form from any of the supported chemical drawing packages.
Assumptions:	Imported chemical structures are assumed to be valid.

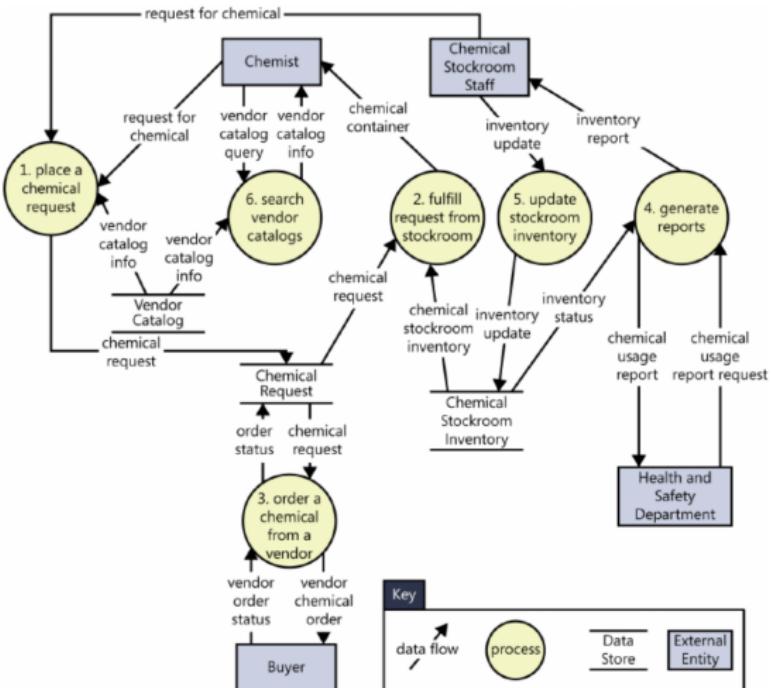
Диаграммы активности



© <https://www.uml-diagrams.org/>

Data-flow diagrams

- ▶ сущности системы
- ▶ внешние сущности
- ▶ потоки данных
- ▶ иерархическая модель



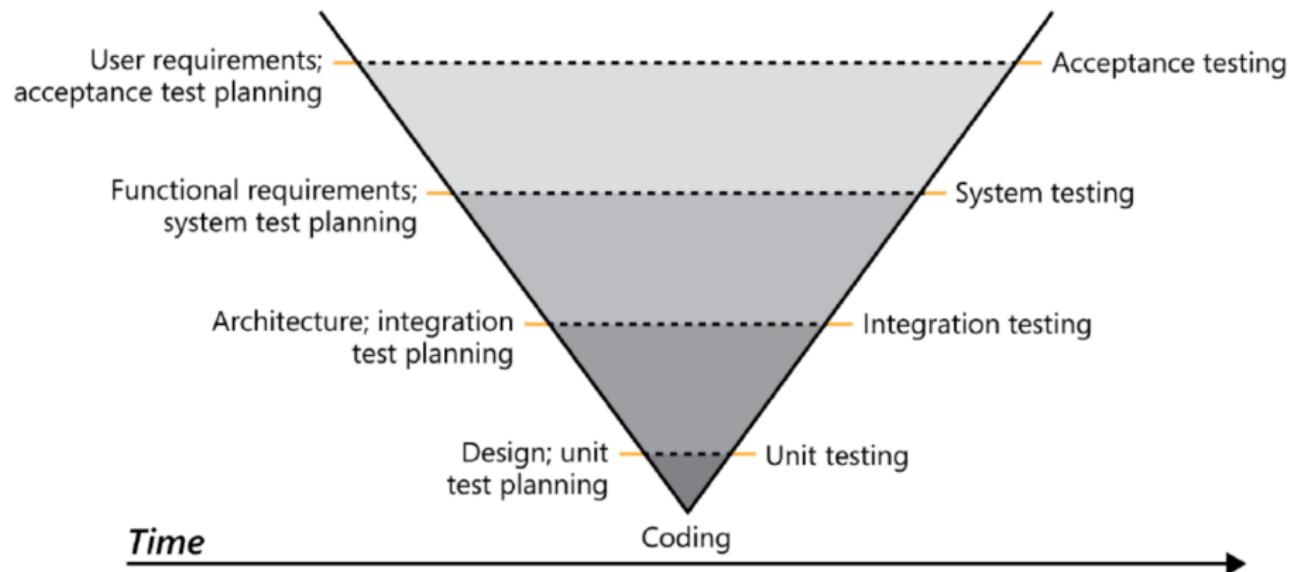
Спецификация требований к ПО (SRS) (1)

- ▶ Введение
 - ▶ Цели
 - ▶ Соглашения о терминах
 - ▶ Предполагаемая аудитория
 - ▶ Масштаб проекта
 - ▶ Ссылки на источники
- ▶ Общее описание
 - ▶ Видение продукта
 - ▶ Функциональность продукта
 - ▶ Классы и характеристики пользователей
 - ▶ Среда функционирования продукта
 - ▶ Рамки, ограничения, правила и стандарты
 - ▶ Документация для пользователей
 - ▶ Допущения и зависимости

Спецификация требований к ПО (SRS) (2)

- ▶ Функциональность системы
 - ▶ Функциональный блок X
 - ▶ Описание и приоритет
 - ▶ Причинно-следственные связи, алгоритмы
 - ▶ Функциональные требования
- ▶ Требования к внешним интерфейсам
 - ▶ Интерфейсы пользователя (UX)
 - ▶ Программные интерфейсы
 - ▶ Интерфейсы оборудования
 - ▶ Интерфейсы связи и коммуникации
- ▶ Нефункциональные требования
 - ▶ Требования к производительности
 - ▶ Требования к сохранности (данных)
 - ▶ Критерии качества ПО
 - ▶ Требования к безопасности системы
- ▶ Прочие требования

Требования и тестирование



Управление требованиями

- ▶ Определение процесса управления изменениями
- ▶ Создание базовой версии и управление версиями требований
- ▶ Использование средств версионирования и управления требованиями
- ▶ Анализ влияния изменений требований
- ▶ Оценка изменяемости требований



Не соглашайтесь, будьте как Уолли

DILBERT



BY SCOTT ADAMS



Атрибуты требований

- ▶ Дата создания, автор
- ▶ Номер текущей ревизии
- ▶ Лицо, ответственное за удовлетворение требования
- ▶ Происхождение или источник требования
- ▶ Состояние (proposed, approved, implemented, verified, rejected и т.п.)
- ▶ Приоритет
- ▶ Подсистемы, для которых актуально требование
- ▶ Номер версии продукта, для которой актуально требование
- ▶ Критерий приемлемости, используемый метод проверки

Доп. информация

<https://stepik.org/course/1128/>

