

## КОНСПЕКТ ЛЕКЦИЙ

по дисциплине

Проектирование человеко-машинного интерфейса

(наименование дисциплины)

Тема лекционных занятий: 2-3.

Бизнес-анализ и обследование объекта автоматизации. Требования к системе

### ***Литература:***

1. IBM(R) Rational Unified Process(R) [Электронный ресурс]: Rational Method Composer - URL: [http://dit.isuct.ru/Publish\\_RUP/](http://dit.isuct.ru/Publish_RUP/)
2. Артефакт: Видение [Электронный ресурс]: Rational Method Composer - URL: [http://dit.isuct.ru/Publish\\_RUP/#core.base\\_rup/workproducts/rup\\_vision\\_2D6D6F1.html](http://dit.isuct.ru/Publish_RUP/#core.base_rup/workproducts/rup_vision_2D6D6F1.html)
3. Вигерс, Карл И. Разработка требований к программному обеспечению. Практические приемы сбора требований и управления ими при разработке программного продукта / Карл И. Вигерс - М.: ИТД "Русская Редакция", 2004
4. BABOK

### ***Краткое содержание лекций по теме:***

1. Аналитик как профессия
2. Методологии разработки ПО
3. Методологии проектирования
4. RUP
5. Разработка документа «Видение»
6. Требования к системе

### **Аналитик как профессия [4]**

Business analyst / бизнес-аналитик - это любой человек, выполняющий задачи бизнес-анализа. Бизнес-аналитики отвечают за выявление, синтез и анализ информации, полученной из различных источников в рамках предприятия, включая инструменты, процессы, документацию и заинтересованные стороны. Бизнес-аналитик отвечает за выявление истинных потребностей заинтересованных сторон, что часто включает исследование и прояснение высказанных ими пожеланий с целью определения лежащих в их основе проблем и причин.

Другие распространенные названия профессий тех, кто выполняет бизнес-анализ, включают:

- бизнес-архитектор,
- аналитик бизнес-систем,
- аналитик данных,
- аналитик предприятия,
- управленческий консультант,

- процессный аналитик,
- менеджер продукта,
- владелец продукта,
- инженер по требованиям и
- системный аналитик.

**ОК 010-2014 (МСКЗ-08). Общероссийский классификатор занятий** (принят и введен в действие Приказом Росстандарта от 12.12.2014 N 2020-ст) (ред. от 18.02.2021), 2511

Системные аналитики - проводят исследования, анализируют и оценивают потребности клиентов в информационных технологиях, процедуры или проблемы клиентов, а также разрабатывают и претворяют в жизнь предложения, рекомендации и планы по улучшению текущих или будущих информационных систем.

К данной группе также относятся следующие занятия:

- аналитик информационных систем,
- бизнес-аналитик (информационные системы),
- консультант по системам,
- разработчик систем (ИКТ),
- специалист в области теории вычислительных машин и систем.

В таблице ниже роли означают следующее:

**Бизнес-аналитик (б/а)** – это аналитик, который хорошо знает предметную область, или детально ее изучает. В его задачу также входит взаимодействие с заказчиками или заинтересованными лицами (далее - stakeholders), клиентами и пользователями разрабатываемой системы; составление и сопровождение бизнес-требований.

**Системный аналитик (с/а)** – это аналитик, который на основе бизнес-требований составляет детализированные и формализованные требования для передачи из программистам (специалистам) в разработку.

Под **Системой** – понимается «Программное обеспечение».

№	Функция (обязанность, действие)	Роль	Примечание
1.	Анализ исходных требований (stakeholder requirements, технического задания) на предмет их корректности и возможности реализации.	б/а, с/а	
2.	Согласование (при необходимости) со stakeholders изменений или дополнений исходных требований.	б/а, с/а	
3.	Сбор исходных данных для проектирования Системы.	б/а, с/а	Включая сохранение собранных исходных данных, их систематизацию и ведение их реестра.
4.	Изучение и описание систем-аналогов.	б/а, с/а	
5.	Разработка модели или описание предметной области.	б/а	

6.	Разработка концепции продукта, границ проекта.	б/а	
7.	Составление (выявление) бизнес-требований, use cases высшего уровня.	б/а	
8.	Составление и проведение презентаций.	б/а	
9.	Составление концепции IT решения (платформа, технологии, архитектура, интерфейсы взаимодействия и так далее).	с/а	Совместно с архитектором.
10.	Составление (выявление) требований к Системе.	б/а, с/а	Основная роль – с/а
11.	Анализ разработок сторонних организаций на предмет их использования в Системе.	б/а, с/а	
12.	Описание структур данных, построение информационной модели.	с/а	
13.	Разработка или участие в разработке протоколов взаимодействия с другими системами, компонент и/или модулей системы между собой.	с/а	Совместно с архитектором.
14.	Разработка сценариев использования (Use Cases), сценариев работы пользователя (User stories).	б/а, с/а	Основная роль – б/а
15.	Участие в разработке GUI.	б/а	Сам GUI разрабатывается UX-дизайнером.
16.	Участие в разработке архитектуры Системы.	с/а	Сама архитектура разрабатывается архитектором.
17.	Участие в разработке прототипа	б/а, с/а	Основная роль – б/а.
18.	Участие программировании (кодировании) Системы.	б/а, с/а	Основная роль – с/а
19.	Валидация Системы.	б/а	
20.	Участие в верификации (тестировании) Системы.	б/а, с/а	Основная роль – с/а
21.	Участие в разработке документации, ее валидация и проверка документации.	б/а, с/а	
22.	Разработка программы и методики испытаний	б/а, с/а	
23.	Участие в испытаниях Системы.	б/а, с/а	
24.	Участие во внедрении Системы, включая обучение пользователей.		Как правило, это означает консультирование группы внедрения и сопровождения.
25.	Участие в сопровождении Системы.	б/а, с/а	В-основном, это анализ запросов на изменения (включая выявленные при

			эксплуатации ошибки).
26.	Наполнение базы знаний.	б/а, с/а	

Знания должны соответствовать разрабатываем продуктам, используемым технологиям, специфике работы компании.

№	Знание	Роль	Примечание
1.	Специфика предметной области (космос, медицина, образование, финансы и так далее).	б/а	
2.	Бизнес-правила (законы, нормы и так далее).	б/а	
3.	Образ пользователя разрабатываемой ИТ-системы/программного продукта (далее – Системы).	б/а	
4.	Стандарты и регламенты составления технической и проектной документации (ГОСТ, ISO/IEC/IEEE и так далее).	б/а, с/а	Достаточно иметь общее представление.
5.	Базовые принципы и концепции структурного и объектно-ориентированного программирования.	с/а	
6.	Язык программирования.	с/а	Достаточно базовых знаний хотя бы одного языка.
7.	Виды тестирования.	с/а	
8.	Технология разработки (создания) Систем.	б/а, с/а	б/а - на бизнес-уровне, с/а – на системном.
9.	Специфика работы Системы на соответствующей платформе/технологии (Web, мобильные приложения, desktop приложения и так далее).	б/а, с/а	
10.	Методики и инструменты прототипирования.	б/а, с/а	Достаточно иметь общее представление.
11.	Протоколы взаимодействия Систем.	с/а	
12.	Основы баз данных.	б/а, с/а	
13.	Основы SQL.	с/а	
14.	Методы и нотации моделирования (Use-Case, Data Flow Diagram и так далее).	б/а, с/а	
15.	Основные цели, задачи и принципы управления требованиями на всех стадиях жизненного цикла Системы.	б/а, с/а	
16.	Атрибуты качества программного обеспечения.	б/а, с/а	Достаточно приводить по памяти 2-3. Все атрибуты должны быть перечислены в соответствующих

			главах шаблона SyRS и SRS.
17.	Атрибуты качества требований.	б/а, с/а	Достаточно приводить по памяти 2-3. Все атрибуты должны быть перечислены в рабочей инструкции.
18.	Виды нефункциональных требований к Системам.	б/а, с/а	Достаточно приводить по памяти 2-3. Все требования должны быть перечислены в соответствующих главах шаблона SyRS и SRS.
19.	Структуры данных и способы их описания.	с/а	
20.	Основы бизнес-анализа (BAWOK) и его методов (техник), включая технику выявления требований.	б/а, с/а	
21.	Основы управления рисками.	б/а, с/а	
22.	Виды испытаний Систем.	б/а, с/а	
23.	Основные методологии разработки Системы.	б/а, с/а	

Умения аналитика:

№	Умение	Роль	Примечание
1.	Пользоваться инструментами для доступа к данным и извлечения данных из реляционных баз данных.	с/а	
2.	Читать исходный программный код.	с/а	В случае крайней необходимости и только для знакомых языков.
3.	Выявлять требования.	б/а, с/а	
4.	Управлять беседой при интервью.	б/а, с/а	
5.	Организовывать взаимодействие с заказчиками и пользователями.	б/а	
6.	Осуществлять декомпозицию требований.	б/а, с/а	
7.	Демонстрировать реализацию функций заинтересованным лицам.	б/а, с/а	
8.	Вести деловую переписку.	б/а, с/а	
9.	Вести деловые переговоры.	б/а, с/а	

10.	Составлять сценарии работы пользователя.	б/а	
11.	Описывать алгоритмы.	с/а	
12.	Описывать структуры данных.	с/а	
13.	Оформлять презентации, схемы и иллюстрации.	б/а, с/а	
14.	Пользоваться системами управления задачами, запросами и дефектами.	б/а, с/а	
15.	Пользоваться системами управления знаниями.	б/а, с/а	
16.	Разрабатывать эскизы интерфейса пользователя.	б/а, с/а	
17.	Обосновывать проектные решения и требования.	с/а	
18.	Описывать организационную, техническую, функциональную, информационную структуры Системы.	с/а	
19.	Моделировать алгоритмы поведения и взаимодействия Системы с окружением.	б/а, с/а	
20.	Формулировать требования.	б/а, с/а	
21.	Производить анализ влияния изменений.	б/а, с/а	
22.	Работать с претензиями потребителей (пользователей).	б/а, с/а	
23.	Искать способы решения проблем пользователей.	б/а, с/а	
24.	Пользоваться средствами прототипирования пользовательских интерфейсов.	б/а, с/а	
25.	Организовывать экспертную оценку предложений.	б/а, с/а	
26.	Разрабатывать деление на подсистемы, этапность и очередность разработки Системы.	б/а, с/а	
27.	Проводить совещания.	б/а, с/а	
28.	Обосновывать выбранные и разработанные методы и шаблоны.	б/а, с/а	

### Методологии разработки ПО

- Каскадная (водопадная) модель (W.W. Royce, 1970 г.)
- Инкрементальная модель
- Спиральная модель (Б. Бозм, 1988 г.)

### Методологии проектирования

Методология проектирования ИТ-систем – набор стандартизованных и апробированных действий, которые позволяют достичь запланированных функциональностей ИТ-систем средствами имеющихся технологий с учетом заданных ограничений. Применение методологии гарантирует упорядоченный подход к промышленной разработке, использованию и сопровождению ИТ-систем, т.е. вносят в процесс создания ПО инженерный подход.

Набор методологий, используемых при проектировании ИТ-систем, достаточно широк. Наиболее распространенные методологии представлены в нижеследующем списке:

- Как получится (code&fix)
- Cleanroom Software Engineering
- Итеративная
  - RUP

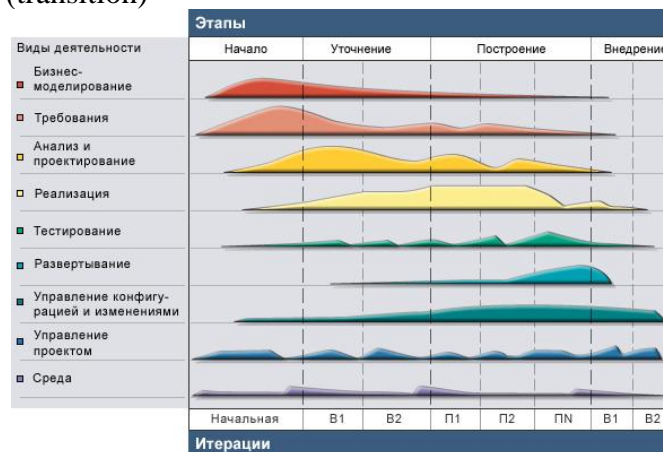
- OpenUP
- MSF
- RAD
- Agile
  - Agile Modeling
  - Agile Unified Process (AUP)
  - Agile Data Method
  - DSDM
  - Essential Unified Process (EssUP)
  - Extreme programming, XP
  - Feature Driven Development (FDD)
  - Getting Real
  - Open Unified Process (OpenUP)
  - Scrum
  - Бережливая разработка программного обеспечения (Lean Software Development)
  - КанБан
  - FDD

## RUP

В качестве примеров реализации итеративного подхода ниже рассматривается методология разработки программного обеспечения, созданные компанией Rational Software – Rational Unified Process (RUP).

RUP – это методология создания программного обеспечения, оформленная в виде размещаемой на Web базы знаний, которая снабжена поисковой системой [1]. RUP использует итеративную модель разработки. Полный жизненный цикл разработки продукта состоит из четырех фаз, каждая из которых включает в себя одну или несколько итераций:

- Начало (inception)
- Уточнение, проектирование (elaboration)
- Построение, конструирование (construction)
- Внедрение (transition)



В матрице RUP итерации (время) расположены по вертикали, а по горизонтали – рабочие процессы (workflows), которые разделены на рабочие процессы процесса (core process workflows)

- бизнес-моделирование
- требования
- анализ и проектирование
- реализация (implementation – выполнение)
- тестирование
- развертывание (deployment) и рабочие процессы поддержки (core supporting workflows)
- управление конфигурациями
- управление проектом
- взаимодействие с окружением

### Разработка документа «Видение»

Документ «Видение» является артефактом бизнес-моделирования и создается на этапе Начала. Начало (inception) подразумевает предварительное взаимодействие с заинтересованными лицами (заказчик, пользователи, инвесторы и др. stakeholders). На этом этапе формируются видение и границы проекта, создается экономическое обоснование, определяются основные требования, ограничения и ключевая функциональность продукта, создается базовая версия модели прецедентов, оцениваются риски [2].

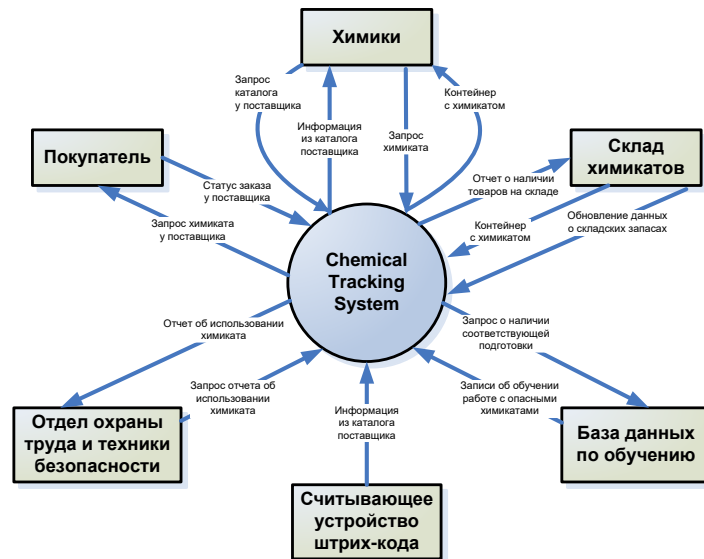
Какие навыки прокачиваются-используются на этом этапе:

Функция	Анализ исходных требований (stakeholder requirements, технического задания) на предмет их корректности и возможности реализации.	б/а, с/а
Функция	Согласование (при необходимости) со stakeholders изменений или дополнений исходных требований.	б/а, с/а
Функция	Сбор исходных данных для проектирования Системы.	б/а, с/а
Функция	Изучение и описание систем-аналогов.	б/а, с/а
Функция	Разработка концепции продукта, границ проекта.	б/а
Функция	Составление (выявление) бизнес-требований, use cases высшего уровня.	б/а
Функция	Составление (выявление) требований к Системе.	б/а, с/а
Знание	Специфика предметной области (космос, медицина, образование, финансы и так далее).	б/а
Знание	Бизнес-правила (законы, нормы и так далее).	б/а
Знание	Образ пользователя разрабатываемой ИТ-системы/программного продукта (далее – Системы).	б/а
Знание	Стандарты и регламенты составления технической и проектной документации (ГОСТ, ISO/IEC/IEEE и так далее).	б/а, с/а
Знание	Специфика работы Системы на соответствующей платформе/технологии (Web, мобильные приложения, desktop приложения и так далее).	б/а, с/а
Знание	Атрибуты качества программного обеспечения.	б/а, с/а
Знание	Атрибуты качества требований.	б/а, с/а



Знание	Виды нефункциональных требований к Системам.	б/а, с/а
Знание	Основные методологии разработки Системы.	б/а, с/а
Умение	Выявлять требования.	б/а, с/а
Умение	Управлять беседой при интервью.	б/а, с/а
Умение	Организовывать взаимодействие с заказчиками и пользователями.	б/а
Умение	Осуществлять декомпозицию требований.	б/а, с/а

- Использование подходов и методологий:
  - мозговой штурм
  - 5WHY
  - SWOT
  - черный ящик
  - функциональное моделирование (IDEF0)
  - контекстная диаграмма
  - майндкарта
- Работа с документами:
  - Изучение
  - Деловая переписка
  - Ведение документов
- Коммуникативные навыки [3, см. главы 6-7]:
  - Фасилитатор - направляет разговор
  - Медиатор - уравнивает противоположные мнения до решения, которое будет выгодно всем
  - Интервьюирование – точные вопросы (да\нет, выбор одного из вариантов ответа, приоритизация)
- Работа с требованиями [2], [3, см. глава 5, приложение Г Документ об образе и границах проекта]:
  - Целеполагание (позиционирование) – необходима правильная постановка проблемы (задачи). *Заказчик (ребенок) хочет Луну. Вы строите космический корабль и отправляете человека на Луну. Однако Заказчик плачет - ребенок хотел игрушку.* В том числе необходимо определить, в чем заключается конкурентоспособность разрабатываемого продукта
  - Выявление заинтересованных лиц – знакомство и сбор контактных данных. Определение степени влияния на продукт каждого заинтересованного лица, их потребности и степень заинтересованности.
  - Определение границ системы – определение зон ответственности (и где эта зона заканчивается), выявление интерфейсов аппаратного, программного, пользовательского взаимодействий, фильтрация требований в связи с установленными границами системы. Границы системы могут быть представлены контекстной диаграммой.



- Выявление требований

## Требования к системе

Требование – это условие или возможность, которой должна соответствовать система [1].

Требование – это: 1. условия или возможности, необходимые пользователю для решения проблем или достижения целей; 2. условия или возможности, которыми должна обладать система или системные компоненты, чтобы выполнить контракт или удовлетворять стандартам, спецификациям или другим формальным документам; 3. документированное представление условий или возможностей для пунктов 1 и 2. (IEEE Standard Glossary of Software Engineering Terminology).

Типы требований [3, см. глава 1]:

1. Функциональные
  - a. Бизнес-требования
  - b. Пользовательские требования
  - c. Функциональные требования
  - d. Системные требования
2. Нефункциональные
  - a. Бизнес-правила
  - b. Атрибуты качества
  - c. Внешний интерфейс
  - d. Ограничения

Тип	Определение
Бизнес-требования	Высокоуровневые цели организации или заказчиков системы. Вся информация, описывающая финансовые, рыночные или другие отношения коммерческого характера, которые клиенты или разработчики собираются получить от использования продукта
Пользовательские	Описывают цели и задачи, которые пользователям позволит решить система
Функциональные	Функциональность ПО, которую разработчики должны построить, чтобы пользователи смогли выполнить свои задачи в рамках бизнес-требований
Системные	Высокоуровневые требования к продукту, которые содержат многие подсистемы, то есть система
Бизнес правила	Бизнес-правило - это положение, определяющее или ограничивающее какие-либо стороны бизнеса; его назначение — защитить структуру бизнеса, контролировать или влиять на его операции
Атрибуты качества	Дополнительное описание функций продукта, выраженное через описание его характеристик, важных для пользователей или разработчиков
Внешний интерфейс	Требования к внешнему виду пользовательского интерфейса и формам взаимодействия с пользователем; Требования по доступу к внутренней функциональности системы при помощи пользовательского интерфейса. Другими словами, первая группа требований описывает взаимодействие подсистемы интерфейса с пользователем, а вторая - с внутренней логикой системы.
Ограничения	Касаются выбора возможности разработки внешнего вида и структуры продукта

Требования к требованиям:

- Атомарность
- Описание только того, что система «должна» (shall) делать, избегать отрицания, двойные отрицания
- Помнить о default, «обычно», «все так делают» (login = email)
- Естественный язык, без жаргона и излишних сокращений
- Корректные
- Однозначные (недвусмысленные)
- Полные, но без излишней детализации (CRUD)
- Последовательные (непротиворечивые)
- Выполнимые
- Ранжированы по приоритетам и/или постоянству
- Проверяемые/Тестируемые («ловушка с ГОСТ 34»)
- Прослеживаемые (трассировка требований)

- Принадлежность к иерархии
- Избегать включения в требования деталей GUI решений, архитектурного дизайна

*Нормативные документы по работе с требованиями:*

Разработки IEEE:

- IEEE 1362 “Concept of Operations Document”.
- 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) – SWEBOK®.

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.

Книги

- Вигерс К. Разработка требований к программному обеспечению/Пер, с англ. — М.:Издательско-торговый дом «Русская Редакция», 2004. —576с.: ил.
- Леффингуелл Д., Уидриг Д. Принципы работы с требованиями к программному обеспечению. М.: ИД “Вильямс”, 2002.
- Коберн А. Современные методы описания функциональных требований к системам. М.: издательство «Лори», 2002. – 263 с.
- Мацяшек Л. Анализ требований и проектирование систем. Разработка информационных :с Диалектика-Вильямс
- Орлик С., Булуй Ю. Введение в программную инженерию и управление жизненным циклом ПО Программная инженерия. Программные требования

Разработал:

Яцык А.А.

---

«\_\_\_» \_\_\_\_\_ 20\_\_ г.

