Результат деятельности каждой организации можно измерить в трех показателях — прибыль, капитализация и реноме. Если деятельность предприятия не направлена на улучшение этих показателей, то ее генеральный директор управляет им неэффективно.

Бизнес-стратегия часто разрабатывается по нескольким направлениям:

* стратегии роста (продуктовая, повышение эффективности, капитализация, партнерство, построение сети)
* инновационные стратегии
* конкурентные стратегии («гонка за лидером», дифференциации, лидерства в издержках, «война стандартов», выживания, поглощения)

Информатизация — комплекс мероприятий, направленных на то, чтобы модель

бизнеса компании наполнялась с помощью ИТ реальным информационным

содержанием и получала долгосрочные конкурентные преимущества.

Архитектура ИТ — это согласованная система представлений об организации

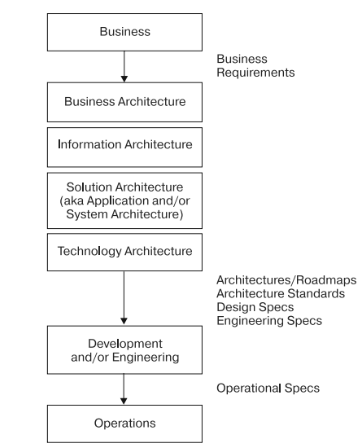
информационной среды компании. Данная система должна соответствовать модели бизнеса, определять целевое состояние ИТ, быть основой долгосрочного планирования, обеспечиваться процессами и инфраструктурой ИТ. Совокупность модели бизнеса и архитектуры ИТ представляет собой корпоративную архитектуру.

Корпоративное лидерство — сочетание трех важных факторов:

* эффективной организации
* бизнес-синергии: включает в себя создание синергии между корпоративными стратегиями, ресурсами и способностями, заполнение «белых пятен» между бизнес-единицами и дивизионами, создание эффективных связей между ними.
* менеджерской команды: создание здоровой команды корпоративных лидеров и поддержание ее эффективности на высоком уровне, выращивание новых лидеров, создание системы мотивации сотрудников, наблюдение затем, чтобы только лучшие люди принимались на работу, развивались дальше, вознаграждались и росли, вкладывая в бизнес свои лучшие силы и способности.

«Архитектура — это базовая организация системы, воплощенная в ее компонентах, их отношениях между собой и с окружением, а также принципы, определяющие проектирование и развитие системы».

Архитектура предприятия (корпоративная архитектура) — это высокоуровневая архитектура всего предприятия, отражающая бизнес-потребности компании, реализуется с помощью ИТ-средств. Фокусируется на определении потоков и бизнес-процессов, действий, функций, информации, данных и технологий предприятия, а также на вызовах, стоящих перед ИТ.

**Business Architecture.** Описывает все бизнес-процессы, бизнес-факторы, бизнес-сущности и бизнес-правила с точки зрения бизнеса. Business Architecture не зависит от применяемых ИТ-технологий.

**Information Architecture.** Определяет структуры данных и описывает все потоки данных, которые используются для поддержки Business Architecture. Такие операции, как идентификация, систематизация, категоризация, хранение данных, относятся к Information Architecture. Может представляться в виде Data Model.

**Solution (System/Application) Architecture.** Архитектура программного обеспечения, которое реализует функции Business Architecture.

**Technology Architecture.** Описывает архитектуру ИT-окружения, которое ис-

пользуется для поддержки Information Architecture и Solution (System/Applica-

tion) Architecture.

**Data Architecture.** Составная часть System Architecture. Описывает структуры

данных и логические связи между данными (hierarchical model, network model,

relational model, object model и т. д.).

Под требованием заказчика понимается потребность или ожидание, которое:

* установлено;
* предполагается (подразумевается);
* является обязательным.

Рассмотрим это на примере. В беседе заказчик выразил ожидание, которое

звучит так: «Сделайте мне редактор писем, чтобы я мог выделять разные слова

разным цветом». В данном случае:

* установлено: программа должна уметь читать формат писем, редактировать письма и сохранять их;
* подразумевалось: «естественно, я хочу, чтобы это был встроенный в почтовый клиент редактор» (в идеале должно быть установлено, а значит, сформулировано аналитиком явно);
* обязательно: выделять разные слова разным цветом.

Приведенный пример показывает, что недостаточно просто зафиксировать пожелание заказчика так, как он его высказал. Необходимо провести анализ и добиться того, чтобы зафиксированная информация обладала следующими основными характеристиками требования.

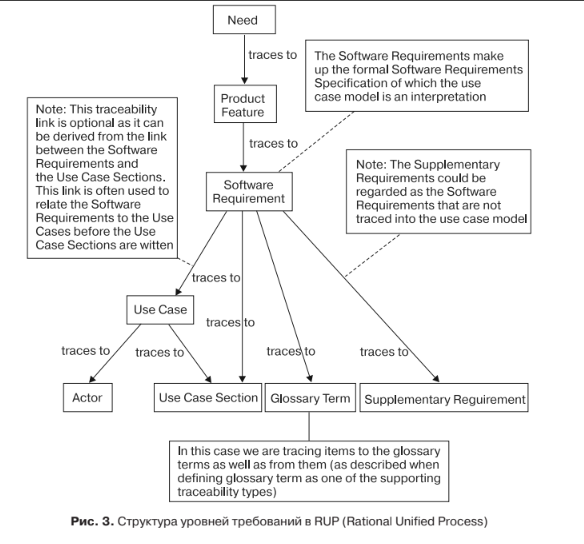
Требование должно быть **тестируемым.**

Требование должно быть **реализуемым.**

Требование должно быть **ясным для понимания.**

Требование должно быть **законченным и полным.**

**Структура уровней и методов в RUP**



| **Квалификация** | **Необходимая теория** | **Необходимые навыки** |
| --- | --- | --- |
| **Младший аналитик** | Книга *«Managing Software Requirements»*  *Dean Leffingwell, Don Widrig*, - я называю ее библия аналитика. Помимо теоретической информации вы найдете в этой книге ряд конкретных примеров и приемов проведения интервью, мозговых штурмов и т.д.  Еще одна – более поздняя и не менее евангелистская книга в области анализа - *Карл И. Вигерс "Разработка требований к программному обеспечению"*  Иметь общее представление о различных методологиях разработки ПО  Знать основы RUP – а именно дисциплину Requirements.  Знать и уметь применять UML, - а именно use case модель, уметь строить domain object model  Понимать основные принципы объектно-ориентированного проектирования и моделирования  Рекомендую также почитать статьи относительно анализа на сайте [www.interface.ru](http://www.interface.ru) | - Выявлять ЗЛ  - Управлять ожиданиями ЗЛ (заинтересованными лицами)  - Проводить собрания  - Проводить интервьюирование  - Проводить анкетирование  - Проводить мозговые штурмы  - Уметь определять границы системы  - Уметь выделять подсистемы и определять их границы  - Уметь собирать и обрабатывать информацию:  *Ответы и собранная информация*  *Запросы заинтересованных лиц*  *Создавать глоссарий и согласовывать термины с ЗЛ*  *Учитывать требования стандартов и ГОСТов при анализе*  *Уметь выявлять характеристики аналогичных / наследуемых систем*  - Уметь выявлять высокоуровневые требования и увязывать их с собранной информацией и между собой:  *Бизнес-требования*  *Бизнес-правила*  *Ограничения и допущения*  *Пользовательские требования*  *Функциональные требования*  *-* Уметь мыслить концептуально и создавать  *Концепцию создания и развития продукта*  *Концепцию системы*  учитывающие выявленные требования  - Проводить основную аналитическую работу по созданию и проектированию системы :  *Уметь проектировать поведение системы и описывать его через требуемые функции системы / варианты использования / прецеденты (use cases)*  *Выявлять нефункциональные требования*  *Требования к пользовательскому интерфейсу*  *Требования к взаимодействию с внешними системами*  - Понимать основные принципы тестирования  - Знать английский язык на уровне, достаточном для чтения технической литературы |
| **Аналитик** | Все теоретические знания младшего аналитика, а также:  Знать и уметь применять ГОСТ 34 и 19 серий  Знать и уметь применять нотацию IDEF0, диаграммы eEPC (extended Event Process Chain)  Книги:  *«Use Case Driven Object Modeling with UML: A Practical Approach.» Doug Rosenberg, Kendall Scott.*  *«Agile Development with ICONIX Process» Doug Rosenberg*  Знать основы RUP – а именно дисциплины Requirements, Analysis & Design.  Знать и уметь строить robustness и sequence diagrams, analysis model  Расширять свой кругозор в методологиях. Познакомиться с гибкими методологиями, особенно рекомендую познакомиться с Iconix. (<http://iconixprocess.com> )  Рекомендую регулярно посещать сайты [www.uml2.ru](http://www.uml2.ru), [www.it4business.ru](http://www.it4business.ru), [www.agilerussia.ru/](http://www.agilerussia.ru/) | Все навыки младшего аналитика, а также:  - Знать, что такое План управления требованиями и уметь его разрабатывать  - Понимать какие модели существуют и где их место в разработке ПО  - Уметь создавать модель анализа  - Строить robustness и sequence диаграммы, понимать, зачем их надо строить и что это дает  - Уметь читать программный код  - Иметь навыки проведения презентаций |
| **Старший / ведущий аналитик** | Все теоретические знания аналитика, а также:  Книги:  *«Архитектура и Стратегия. "Инь" и "Янь" информационных технологий предприятия.»*  *Данилин А., Слюсаренко А.*  *«A Spiral Model of Software Development and Enhancement".»*  *Boehm Barry W.*  *«A Guide to the Project Management Body of Knowledge»*  *ANSI / PMI*  Рекомендую расширять свои знания методологий разработки программного обеспечения, обязательно рекомендую обратить внимание на <http://ru.wikipedia.org/wiki/Спиральная_модель>  Статья*:*  *Кристофер Эберт «Understanding the Product Life Cycle: Four Key Requirements Engineering Techniques»*  Рекомендую регулярно посещать <http://msdn.microsoft.com/ru-ru/default.aspx> и <http://www-01.ibm.com/software/success/cssdb.nsf/topstoriesFM?OpenForm&Site=rational&cty=en_us> | Все навыки аналитика, а также:  - Иметь детальное представление о ЖЦ (жизненном цикле) проекта и продукта  - Знать, что такое План управления документами и уметь его создавать  - Уметь создавать логическую модель и модель данных  - Желательно уметь создавать простой программный код (этот навык нужен для закрепления навыка уметь читать программный код, никто так не поймет разработчика, как другой разработчик, и, если вы хотите говорить с разработчиками на одном языке, вам придется освоить программирование в минимальном объеме.)  - Уметь профессионально проводить презентации  - Проводить выученные уроки по практикам разработки и управления требованиями  - Быть наставником для аналитиков  - Уметь предотвращать и разрешать конфликты в проектной команде  - Уметь выявлять и управлять рисками |
|  |  |  |

**Младший аналитик**

**Процесс обработки и анализа информации**

Процесс обработки и анализа информации можно представить в виде следующих шагов:

* определить группы/категории информации;
* определить атрибуты для дополнительной категоризации информации;
* разбить информацию на определенные ранее группы;
* внутри каждой группы выделить главные информационные составляющие и построить от них иерархию;
* присвоить значения дополнительным атрибутам, определенным ранее;
* определить перечень возможных состояний информационных составляющих;
* присвоить состояния информационным составляющим;
* провести трассировки/обеспечить прослеживаемость одной информационной составляющей из другой.