



Основная проектная документация в работе БА.

Роль бизнес-аналитика



Оглавление

Введение	2
Термины, используемые в лекции	3
V&S (Vision & Scope)	3
BRD (Business Requirements Document)	6
SRS (Software Requirements Specification)	10
ТЗ	13
Инструменты для ведения документации	15
Что можно почитать еще?	16

Введение

В процессе разработки любого проекта, в том числе программного обеспечения, существует необходимость в создании и поддержании специальной документации, которая описывает основные аспекты проекта. Она может включать в себя различные типы документов, в том числе V&S (Vision & Scope), BRD (Business Requirements Document), SRS (Software Requirements Specification) и ТЗ (Техническое задание). В данной лекции мы рассмотрим каждый из этих документов и обсудим их значение и применение в работе бизнес-аналитика.

Мы пройдем:

- Различные типы документов, используемых в процессе разработки, включая V&S (Vision & Scope), BRD (Business Requirements Document), SRS (Software Requirements Specification) и ТЗ (Техническое задание);
- Описание и значимость каждого из этих типов документов;
- Практическое применение каждого из документов в работе бизнес-аналитика;
- Примеры использования документов в реальных проектах;
- Инструменты для документации.

Термины, используемые в лекции

V&S (Vision and Scope) – это документ, который описывает видение и область действия проекта. Он разрабатывается в начале проекта и служит основой для всех последующих документов и работ по проекту.

BRD (Business Requirements Document) – это документ, который описывает бизнес-требования к проекту. Он предназначен для того, чтобы дать понимание того, что должно быть разработано или улучшено в проекте, чтобы он отвечал бизнес-потребностям заказчика.

SRS (Software Requirements Specification) – это документ, который описывает требования к разрабатываемому программному обеспечению. Он является основой для разработки, тестирования и внедрения продукта.

Техническое задание (ТЗ) – это документ, который определяет требования к проекту или продукту, который должен быть разработан. ТЗ подробно описывает требования к функциональности, аппаратному и программному обеспечению, а также другим характеристикам, необходимым для успешного выполнения проекта.

V&S (Vision & Scope)

Документ Vision & Scope (V&S) представляет собой краткое описание проекта, его видение и предполагаемую область применения. Он описывает общие цели и направления проекта, а также определяет, какие бизнес-процессы будут затронуты проектом.

Этот документ является важным исходным материалом для разработки других проектных документов и позволяет заинтересованным сторонам оценить ценность и пользу проекта. Документ V&S также может помочь бизнес-аналитикам и проектным менеджерам оценить соответствие проекта бизнес-стратегии компании.

Структура документа Vision & Scope (V&S) может варьироваться в зависимости от требований проекта и специфики бизнеса, но обычно он включает в себя следующие разделы:

1. Введение

Этот раздел содержит общую информацию о проекте, его целях и задачах. Здесь также могут быть указаны ключевые заинтересованные стороны (стейкхолдеры) проекта и описание продукта, который должен быть создан.

2. Бизнес-видение

Этот раздел описывает общее видение продукта, его цели и ценности для пользователей и бизнеса. Здесь также могут быть указаны ключевые функциональные возможности, которые должны быть реализованы.

3. Границы проекта

Этот раздел описывает, какие части продукта будут включены в проект и какие будут исключены. Здесь также могут быть указаны ключевые ограничения и зависимости, которые могут повлиять на проект.

4. Требования к продукту

Этот раздел содержит подробное описание верхнеуровневых функциональных и нефункциональных требований к продукту. Здесь могут быть описаны основные возможные сценарии использования и функции продукта.

5. Риски и возможности

Этот раздел описывает все возможные риски, связанные с проектом / продуктом, и возможности их предотвращения. Здесь также могут быть указаны возможные проблемы и способы их решения.

6. Команда проекта и роли

Этот раздел описывает состав команды проекта и роли каждого участника. Здесь также могут быть указаны ожидания по поводу взаимодействия между участниками команды и другими заинтересованными сторонами проекта.

7. Планирование и бюджет

Этот раздел содержит общее описание планирования проекта, включая временные рамки и бюджет проекта. Здесь также могут быть указаны ключевые мероприятия и этапы проекта, а также описание процесса управления изменениями.

8. Результаты

Ожидаемые результаты от проекта, который планируется реализовать. В этом разделе можно указать конкретные показатели, которые должны быть достигнуты в результате реализации проекта.

Пример Vision & Scope (V&S)

Давайте рассмотрим более подробный пример Vision & Scope (V&S) для проекта создания онлайн-платформы для образования в сфере информационных технологий.

Vision проекта: Создать онлайн-платформу, которая поможет людям повысить свои знания и навыки в области информационных технологий, предоставляя доступ к качественным онлайн-курсам и обучающим материалам.

Цели проекта:

- Создать интуитивно понятный и удобный интерфейс для пользователей;
- Обеспечить доступность обучения для широкой аудитории пользователей;
- Предоставить качественные онлайн-курсы по различным темам в области информационных технологий;
- Обеспечить высокую производительность платформы и быстрый отклик на действия пользователя;
- Создать систему мотивации пользователей для продвижения их обучения;
- Интегрировать платформу с социальными сетями и почтовыми сервисами для удобного обмена материалами и коммуникации между пользователями.

V&S также включает описание ключевых функций платформы, пользовательских сценариев, анализ конкурентов и рынка, технические требования и оценку бюджета проекта.

Описание ключевых функций платформы может включать следующие элементы:

- Регистрация и авторизация пользователей;
- Поиск и выбор курсов;
- Обзор курсов и описание уровня сложности;
- Просмотр видеоуроков и выполнение заданий;
- Отслеживание прогресса и мотивация пользователей;
- Платежная система для оплаты курсов;
- Обратная связь и поддержка пользователей.

Пользовательские сценарии могут включать:

- Регистрация нового пользователя и выбор курса;
- Просмотр видеоуроков и выполнение заданий;
- Отслеживание прогресса и получение мотивации для продвижения обучения;
- Оплата курсов и получение сертификата.

Анализ конкурентов и рынка может включать описание основных конкурентов, рыночного потенциала и потенциальной аудитории.

Технические требования могут включать:

- Разработка платформы на современном языке программирования;
- Использование современных фреймворков для обеспечения высокой производительности платформы;

- Интеграция с базами данных и серверами для обеспечения быстрого доступа к материалам и учета прогресса пользователей;
- Обеспечение безопасности и защиты персональных данных пользователей;
- Разработка мобильных приложений для удобства использования платформы на смартфонах и планшетах;

Оценка бюджета может включать описание затрат на разработку платформы, разработку контента для курсов, маркетинг и рекламу, поддержку и обновление платформы.

Как мы убедились на данном примере, V&S для проекта создания онлайн-платформы для образования в сфере информационных технологий помогает определить цели и задачи проекта, описать ключевые функции и пользовательские сценарии, проанализировать рынок и конкурентов, определить технические требования и оценить бюджет проекта. Это важный документ, который помогает сформировать общее видение проекта и обеспечить его успешную реализацию.

BRD (Business Requirements Document)

Business Requirements Document (BRD) – это документ, описывающий бизнес-требования к проекту. Это один из основных документов, который помогает бизнес-аналитику уточнить и документировать бизнес-требования к проекту, на основе которых затем будет разрабатываться продукт. BRD используется как инструмент коммуникации между бизнес-аналитиком и командой разработки.

Структура документа BRD может незначительно различаться в зависимости от конкретных потребностей проекта и требований заказчика, но обычно включает следующие разделы:

1. **Введение:** в этом разделе кратко описывается проект, его цели и основные задачи.
2. **Описание бизнес-процессов:** здесь описываются процессы, в рамках которых будет использоваться продукт. Также описываются проблемы, с которыми сталкиваются пользователи в этих процессах и которые должны быть решены с помощью продукта.
3. **Требования к продукту:** в этом разделе могут описываться верхнеуровневые требования к продукту, которые необходимы для реализации описанных в предыдущем разделе бизнес-процессов.

4. **Ограничения и предположения:** здесь описываются предположения и ограничения, которые могут повлиять на реализацию проекта.
5. **Риски:** этот раздел описывает возможные риски и препятствия, которые могут возникнуть в процессе реализации проекта.
6. **График работ:** в этом разделе указывается график работ по проекту, включая временные рамки для каждого этапа.
7. **Список участников:** здесь перечисляются все участники проекта и их роли.
8. **Список требований:** это опциональное приложение к документу BRD, которое содержит полный список всех требований к продукту, описанных в разделе "Требования к продукту".

Пример BRD (Business Requirements Document) для разработки онлайн-магазина:

1. Введение

1.1 Цель и описание проекта

- Цель проекта: создание онлайн-магазина для продажи товаров широкого ассортимента;
- Описание проекта: онлайн-магазин должен предоставлять удобный интерфейс для поиска, просмотра и покупки товаров, а также возможность оплаты заказа и доставки товара.

1.2 Бизнес-цели

- Увеличение объема продаж компании;
- Расширение рынка сбыта и увеличение доли рынка;
- Повышение удовлетворенности клиентов.

1.3 Целевая аудитория

- Потенциальные клиенты: люди, заинтересованные в покупке товаров через интернет;
- География: Россия;
- Возраст: от 18 лет.

2. Общее описание проекта

2.1 Верхнеуровневые требования к функционалу

- Создание каталога товаров с возможностью фильтрации и сортировки;
- Регистрация и авторизация пользователей;
- Возможность добавления товаров в корзину;
- Оформление заказа с выбором способа оплаты и доставки;

- Поддержка различных способов оплаты, включая банковские карты и электронные кошельки;
- Возможность отслеживания статуса заказа;
- Отправка уведомлений о статусе заказа клиентам.

2.2 Верхнеуровневые нефункциональные требования

- Производительность: время загрузки страницы не должно превышать 3 секунд;
- Надежность: магазин должен быть доступен для пользователей в течение 99% времени;
- Безопасность: должны быть применены меры для защиты личных данных клиентов и их платежных реквизитов.

3. Требования к инфраструктуре

3.1 Хостинг и серверное оборудование

- Для хранения данных магазина должен использоваться выделенный сервер;
- Серверное оборудование должно иметь достаточную производительность для обработки запросов большого количества пользователей.

3.2 Программное обеспечение

- Для создания онлайн-магазина должен использоваться фреймворк Ruby on Rails;
- Для хранения данных магазина должна использоваться база данных MySQL.

4. Требования к команде

4.1 Команда проекта

- Руководитель проекта;
- Дизайнер;
- Разработчик (Ruby on Rails);
- QA-инженер;
- Системный администратор.

4.2 Требования к навыкам и опыту персонала

- Руководитель проекта должен иметь опыт работы в управлении проектами, знание основных методологий управления проектами (например, Agile);
- Дизайнер должен иметь опыт работы с графическими редакторами и знание принципов дизайна интерфейсов;
- Разработчик должен иметь опыт работы с Ruby on Rails и знание принципов объектно-ориентированного программирования;

- QA-инженер должен иметь опыт работы с тестированием ПО и знание основных методов тестирования;
- Системный администратор должен иметь опыт работы с серверным оборудованием и знание основных принципов администрирования серверов.

5. План работ

5.1 Этапы проекта

1. Анализ требований и подготовка ТЗ;
2. Разработка дизайна;
3. Разработка функционала магазина;
4. Тестирование и отладка;
5. Развертывание на сервере;
6. Запуск магазина.

5.2 Сроки работ

1. Анализ требований и подготовка ТЗ – 4 недели;
2. Разработка дизайна – 2 недели;
3. Разработка функционала магазина – 6 недель;
4. Тестирование и отладка – 2 недели;
5. Развертывание на сервере – 1 неделя;
6. Запуск магазина – 1 день.

6. Бюджет проекта

6.1 Расходы

- Зарплата персонала – 300 000 рублей;
- Хостинг и серверное оборудование – 50 000 рублей;
- Программное обеспечение – 30 000 рублей.

6.2 Планируемая прибыль

- Ожидаемый объем продаж в месяц – 10 000 000 рублей;
- Прибыль в месяц – 2 000 000 рублей.

7. Риски проекта

7.1 Технические риски

- Непредвиденные проблемы с сервером;
- Невозможность поставить необходимое оборудование в срок.

7.2 Бизнес-риски

- Неудачная рекламная кампания;
- Низкая конкурентоспособность ассортимента товаров.

Заключение

Данный BRD описывает основные требования и план работ для проекта создания интернет-магазина. BRD включает в себя информацию о задачах, целях и целевой аудитории проекта, описание основных функциональных возможностей магазина, требования к персоналу, план работ и бюджет проекта, а также описание возможных рисков и мер по их управлению.

BRD используется для согласования и обсуждения проекта между заказчиком и командой проекта, а также в качестве основы для дальнейшей разработки SRS. Он помогает заказчику получить более четкое представление о проекте, а также понимание того, сколько времени и денег потребуется для его реализации.

Пример BRD может быть использован как основа для разработки конкретного проекта. Однако, следует учитывать, что BRD должен быть адаптирован под конкретные потребности и требования заказчика и проекта. Кроме того, BRD может включать в себя дополнительную информацию в зависимости от проекта, например, требования к безопасности или интеграции с другими системами.

SRS (Software Requirements Specification)

Продолжим изучение основных проектных документов в работе бизнес-аналитика и перейдем к **SRS (Software Requirements Specification)** – документу технических требований к программному продукту.

SRS – это документ, описывающий детальные технические требования к разрабатываемому программному продукту. Этот документ является основой для команды разработчиков, поскольку он содержит все необходимые технические детали и спецификации, необходимые для создания продукта.

Пример типичной структуры документа SRS:

1. **Введение:** общее описание системы, ее целей и контекста использования;
2. **Описание продукта:** полное описание системы, ее возможностей, характеристик и ограничений;
3. **Требования к функциональности:** описание детальных функциональных требований, которые должны быть реализованы в системе;
4. **Требования к характеристикам системы:** описание нефункциональных требований, таких как производительность, надежность, безопасность и т.д.;

5. **Требования к интерфейсу:** описание пользовательского интерфейса, включая формы, поля ввода, элементы управления и т.д.;
6. **Требования к базе данных:** описание требований к базе данных, включая ее структуру, ограничения и характеристики;
7. **Требования к аппаратному обеспечению:** описание требований к аппаратному обеспечению, необходимому для работы системы;
8. **Требования к программному обеспечению:** описание требований к программному обеспечению, необходимому для работы системы;
9. **Требования к тестированию:** описание требований к тестированию системы, включая тестовые сценарии, тестовые данные и критерии успешности;
10. **График работ:** описание графика работ по проекту и плана выпуска системы;
11. **Соглашения и стандарты:** описание соглашений и стандартов, используемых в проекте;
12. **Приложения:** дополнительные материалы, такие как диаграммы, таблицы, спецификации и т.д.

Документ SRS является одним из наиболее важных документов в процессе разработки программного обеспечения. Он помогает обеспечить понимание требований к системе всем участникам проекта и служит основой для дальнейшей работы.

Пример SRS может быть следующим:

1. Введение

Цель: Создание программного обеспечения для автоматизации управления складскими запасами.

2. Описание продукта

Программное обеспечение должно предоставлять следующие функции:

- Управление приемом товаров на склад;
- Управление отгрузкой товаров со склада;
- Управление остатками товаров на складе;
- Управление списанием товаров со склада.

3. Требования к продукту

3.1 Требования к функциональности:

- Программное обеспечение должно обеспечивать возможность ввода, хранения и вывода данных по поставкам товаров на склад и отгрузкам товаров со склада;

- Программное обеспечение должно обеспечивать возможность ввода, хранения и вывода данных о текущих остатках товаров на складе;
- Программное обеспечение должно обеспечивать возможность списания товаров со склада с помощью различных методов списания (например, FIFO, LIFO и т.д.);
- Программное обеспечение должно обеспечивать возможность генерации отчетов о приходе, расходе и остатках товаров на складе.

3.2 Требования к надежности:

- Программное обеспечение должно быть надежным и стабильным;
- Программное обеспечение должно быть защищено от несанкционированного доступа.

3.3 Требования к удобству использования:

- Программное обеспечение должно иметь интуитивно понятный интерфейс;
- Программное обеспечение должно быть легко настраиваемым.

3.4 Требования к производительности:

- Программное обеспечение должно обеспечивать быстрый доступ к данным;
- Программное обеспечение должно обеспечивать возможность работы с большим объемом данных.

4. Требования к технической реализации

4.1 Требования к аппаратному обеспечению:

- Система должна работать на сервере с операционной системой Windows Server 2016 или выше;
- Минимальные требования к серверу: процессор Intel Core i5, 8 ГБ ОЗУ, 500 ГБ HDD.

4.2 Требования к программному обеспечению:

- Система должна быть разработана на языке программирования Java;
- Система должна использовать базу данных MySQL версии 8.0 или выше;
- Для создания пользовательского интерфейса должен использоваться фреймворк JavaFX версии 8.0 или выше.

5. Требования к документации

5.1 Техническая документация:

- Должна быть создана документация по установке и настройке программного обеспечения;

- Должна быть создана документация по использованию программного обеспечения.

5.2 Руководство пользователя:

- Должно быть создано руководство пользователя на русском языке;
- Руководство пользователя должно содержать инструкции по установке, настройке и использованию программного обеспечения.

6. Требования к тестированию

6.1 Функциональное тестирование:

- Должно быть проведено функциональное тестирование всех основных функций программного обеспечения.

6.2 Нагрузочное тестирование:

- Должно быть проведено нагрузочное тестирование программного обеспечения для определения максимальной нагрузки, которую система может выдержать.

7. Требования к обеспечению безопасности

7.1 Аутентификация и авторизация:

- Должен быть реализован механизм аутентификации и авторизации пользователей.

7.2 Шифрование:

- Должно быть реализовано шифрование данных, передаваемых по сети.

Мы с вами разобрали пример SRS, который описывает требования к программному обеспечению для автоматизации управления складскими запасами. В зависимости от конкретных требований и условий, SRS может содержать более детальную информацию и дополнительные разделы.

TЗ

TЗ или техническое задание – это документ, который содержит подробное описание требований к созданию или изменению продукта или проекта.

Техническое задание обычно включает в себя следующие разделы:

1. **Введение:** описание общей цели проекта или продукта, а также краткое изложение требований;
2. **Описание продукта:** детальное описание продукта, который должен быть создан или изменен, включая его функциональность, особенности и характеристики;
3. **Требования к проекту:** описание технических, функциональных, производственных и административных требований к проекту;
4. **Требования к разработке:** описание технологий и методов, которые должны использоваться при создании продукта, а также дополнительные требования, такие как тестирование, обучение и поддержка;
5. **Требования к команде:** описание квалификаций, опыта и ресурсов, которые необходимы для создания продукта;
6. **График работ и бюджет:** описание временных рамок и финансовых ресурсов, необходимых для создания продукта.

Техническое задание используется в различных случаях, включая создание новых продуктов, разработку программного обеспечения, создание веб-сайтов и многих других проектов. Оно позволяет уточнить требования заказчика и снизить риски ошибок в процессе разработки. ТЗ также служит основой для контроля качества и проверки, что созданный продукт соответствует требованиям заказчика.

Сокращенный пример технического задания может быть следующим:

1. Введение

Цель проекта: Разработка сайта для компании, занимающейся продажей косметических средств.

2. Описание продукта

Сайт должен содержать следующие разделы:

- Главная страница с описанием продукции и возможностью заказа;
- Каталог продукции с возможностью фильтрации и сортировки;
- Раздел "О нас" с информацией о компании и ее истории;
- Раздел "Контакты" с формой обратной связи и контактной информацией.

3. Требования к проекту

- Сайт должен быть доступен на различных устройствах (настольных компьютерах, планшетах, смартфонах) и иметь адаптивный дизайн;
- Сайт должен быть построен на CMS WordPress;
- Сайт должен быть оптимизирован для SEO;
- Сайт должен поддерживать несколько языков – русский и английский.

4. Требования к разработке

- Сайт должен быть разработан с использованием HTML, CSS, JavaScript и MySQL;
- Дизайн сайта должен быть согласован с заказчиком до начала разработки;
- Сайт должен быть разработан в соответствии с принципами юзабилити;
- Сайт должен поддерживать работу в различных браузерах и операционных системах.

5. Требования к команде проекта

- Разработчик должен иметь опыт работы с CMS WordPress, знание HTML, CSS, JavaScript и MySQL;
- Дизайнер должен иметь опыт работы в сфере веб-дизайна и иметь знание Photoshop и Illustrator.

6. Расписание и бюджет

- Разработка сайта должна занять 2 месяца;
- Бюджет на разработку сайта составляет 100 000 рублей.

Такое ТЗ позволяет разработчикам и дизайнерам точно понимать требования заказчика и создавать продукты, которые полностью соответствуют его ожиданиям.

Инструменты для ведения документации

Существует множество инструментов для ведения документации. Некоторые из них бесплатны, а некоторые требуют подписки или платные лицензии. Ниже приведен список наиболее популярных инструментов для ведения документации и поддержания ее в актуальном состоянии:

- **Confluence**: это один из наиболее популярных инструментов для ведения документации. Он используется для создания страниц, задач, заметок, обсуждений и многого другого. Он интегрируется с другими инструментами Atlassian, такими как Jira и Trello.
- **Notion**: это онлайн-инструмент для ведения заметок, задач, баз данных и документации. Он имеет удобный интерфейс и множество функций.
- **Slite**: онлайн-инструмент для создания и организации документации,

который предлагает функции совместной работы, контроля версий и уведомлений.

- **Nuclino:** платформа для создания и совместного использования знаний, которая может быть использована для документирования процессов, проектов и компонентов.
- **Jira:** это инструмент управления проектами, который также используется для создания и отслеживания задач. Он позволяет создавать задачи для команды и управлять ими, а также комментариями и вопросами к ним.
- **Trello:** это онлайн-доска, на которой можно создавать списки задач, перемещать их между списками, добавлять комментарии, прикреплять файлы и многое другое.
- **Asana:** это онлайн-сервис для управления проектами, который позволяет создавать задачи, присваивать им ответственных, устанавливать сроки выполнения.
- **GitHub:** это популярный инструмент для хранения, управления и совместной работы над исходным кодом. Он также может использоваться для ведения документации.
- **Google Документы:** это бесплатный онлайн-сервис от Google, который позволяет создавать и редактировать текстовые документы, таблицы и презентации. Он также может использоваться для создания и ведения документации.
- **Microsoft SharePoint:** это платформа, которая позволяет создавать и хранить документы, общаться с командой и управлять проектами.

Что можно почитать еще?

1. "Software Requirements" by Karl Wieggers and Joy Beatty – книга, которая покрывает все аспекты процесса сбора и документирования требований.
2. "Writing for Software Developers" by Philip Kiely – книга, которая описывает, как создавать понятную и эффективную документацию для разработчиков.