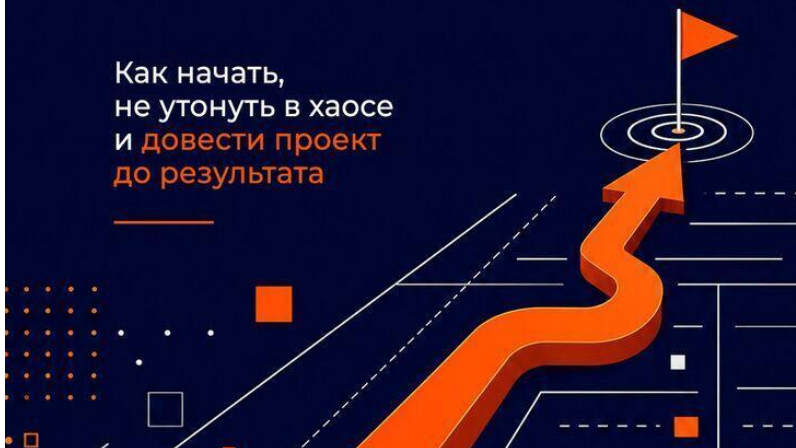


12+

Пётр Дикий

РУКОВОДИТЕЛЬ ПРОЕКТА БЕЗ ИЛЛЮЗИЙ

Как начать,
не утонуть в хаосе
и довести проект
до результата



Петр Дикий
Руководитель проекта
без иллюзий. Как начать,
не утонуть в хаосе и довести
проект до результата

*<https://litres.ru/74275244>
ISBN 9785007069069*

Аннотация

Практическая книга для тех, кто начинает управлять проектами и хочет понять реальную работу РП: как выстроить процесс с нуля, разобраться в требованиях, сроках, ресурсах и рисках, наладить взаимодействие с командой и заказчиком, не утонуть в хаосе и довести результат до приёмки.

Содержание

ПРЕДИСЛОВИЕ	6
ВВЕДЕНИЕ	9
ГЛАВА 1. УПРАВЛЕНИЕ ПРОЕКТАМИ	14
1.1. ЧТО ТАКОЕ ПРОЕКТ?	15
1.2. ПОЧЕМУ УПРАВЛЕНИЕ ПРОЕКТАМИ?	18
1.3. ПРОЕКТНАЯ КОМАНДА И ЕЁ ВАРИАНТЫ	20
1.4. РУКОВОДИТЕЛЬ ПРОЕКТА. КТО ТАКОЙ? КАКИЕ У НЕГО ОБЯЗАННОСТИ?	25
1.5. ЧЕТЫРЕ ЭТАПА УПРАВЛЕНИЯ ПРОЕКТОМ	29
1.5.1. ЭТАП «ИНИЦИАЦИЯ»	33
1.5.2. ЭТАП «ПЛАНИРОВАНИЕ»	36
1.5.3. ЭТАП «ИСПОЛНЕНИЕ»	45
1.5.4. ЭТАП «ЗАКРЫТИЕ»	47
1.6. ВЫБОР МЕТОДОЛОГИИ И РАБОТА С ПОДХОДОМ К ПРОЕКТУ	49
1.6.2. КАК ВЫБРАТЬ?	52
1.6.3. ГИБРИДНЫЕ ПОДХОДЫ: WATERFALL + AGILE	53
1.6.4. ИСТОРИИ УСПЕХА РЕАЛЬНЫХ ПРОЕКТОВ	54

1.7. ЧТО ДЕЛАЕТ ПРОЕКТ УСПЕШНЫМ?	57
1.8. РАБОТА БЕЗ УПРАВЛЕНИЯ	63
ПРОЕКТАМИ. ПОУЧИТЕЛЬНАЯ ИСТОРИЯ	
ГЛАВА 2. ПЛАНИРОВАНИЕ БЕЗ ИЛЛЮЗИЙ:	65
КАК ПРЕВРАТИТЬ ИДЕЮ В УПРАВЛЯЕМЫЙ	
ПРОЕКТ	
2.1. НАЗВАНИЕ ПРОЕКТА	67
2.2. ОПРЕДЕЛЕНИЕ ОБЪЁМА ПРОЕКТА	68
2.3. ОБЪЁМ ПРОЕКТА	69
2.4. УСТАВ ПРОЕКТА	77
2.4.1. ОТ УСТАВА К РЕАЛЬНОСТИ: КАК	85
РАЗЛОЖИТЬ ПРОЕКТ НА ЗАДАЧИ И НЕ	
СОВРАТЬ СРОКАМИ	
2.5. ПЛАНИРОВАНИЕ И ОЦЕНКА	97
РЕСУРСОВ	
2.5.1. ПЛАНИРОВАНИЕ РЕСУРСОВ	98
2.5.2. ОЦЕНКА РЕСУРСОВ	108
2.5.3. ТРЁХТОЧЕЧНЫЙ МЕТОД ОЦЕНКИ	127
ЗАДАЧ	
2.5.4. РЕДЛАЙН: ВНУТРЕННЯЯ КРАСНАЯ	133
ЛИНИЯ СРОКА	
Конец ознакомительного фрагмента.	139

**Руководитель
проекта без иллюзий
Как начать, не утонуть
в хаосе и довести
проект до результата**

Петр Дикий

© Петр Дикий, 2026

ISBN 978-5-0070-6906-9

Создано в интеллектуальной издательской системе Ridero

ПРЕДИСЛОВИЕ

Эта книга — не про идеальный проект из учебника. Она про реальную работу руководителя проекта: когда требования меняются, сроки горят, команда устаёт, заказчик ждёт результата, а хаос сам собой почему-то не исчезает.

Я написал её как практический маршрут для начинающего руководителя проекта: от хаоса на старте до понятных требований, плана, команды, коммуникаций, исполнения, приёмки и закрытия.

Книга основана на моём личном опыте в управлении проектами, командами и изменениями. За годы работы я создавал и запускал проекты для Nissan, Toyota, Альфа-Банка, Burger King, Nestle, Abbott, KT&G, Минпромторга, Росатома и других компаний. Также я обучал руководителей проектов, аналитиков, архитекторов и тимлидов тому, как управлять, планировать, думать системно и доводить работу до результата.

Книга ориентирована в первую очередь на ИТ-проекты, но проектный подход применим гораздо шире: во внутренних изменениях, запуске продуктов, внедрении систем, организационных проектах, сервисных процессах и других задачах, где есть цель, сроки, ограничения, команда и ожидаемый результат.

Эта книга для начинающих и молодых руководителей

проектов, которым нужно спокойно и системно войти в профессию: понять базовые процессы, роли, документы, инструменты, ошибки и управленческую логику. Она будет полезна и опытным специалистам — как настольная книга по базовым вещам, чек-листам, рабочим подходам и формулировкам.

Я не пытаюсь сделать из проектного управления магию. Наоборот, хочу показать его без иллюзий: с хаосом, конфликтами, срывами сроков, слабой постановкой задач, неясными требованиями, ошибками коммуникации, давлением заказчика и ответственностью руководителя проекта за управляемость всей системы.

В книге я простым языком объясняю, как выстраивать управление проектом «с нуля»: как запускать проект, собирать требования, планировать сроки и ресурсы, работать с рисками, выстраивать коммуникации, управлять командой, контролировать исполнение, принимать изменения, доводить результат до приёмки и закрывать проект.

Важно понимать:

Инструменты из книги не нужно внедрять всё сразу. Их нужно адаптировать под масштаб, зрелость и специфику конкретного проекта. В маленьком проекте достаточно нескольких простых инструментов. В крупном — из них можно собрать полноценную систему управления.

Я хочу, чтобы в нашей стране было больше сильных руко-

водителей, которые не просто занимают должность, а понимают свою ответственность, умеют думать, договариваться, принимать решения и доводить проекты до результата.

Буду рад и благодарен любым комментариям. Пишите мне на почту **petr@dikiy.pro** или в социальных сетях.

ВВЕДЕНИЕ

Спасибо, что выбрали эту книгу для погружения в управление проектами или для помощи в текущей работе. К концу книги вы поймёте, как вести проект от идеи до результата, какую роль играет руководитель проекта, какие документы и инструменты нужны на разных этапах и что делать, когда план начинает расходиться с реальностью.

Как читать эту книгу?

Первый раз — последовательно, чтобы увидеть логику управления проектом: от инициации и планирования до исполнения, закрытия и развития руководителя проекта.

Второй раз — как настольную книгу. Возвращайтесь к нужной главе, когда столкнётесь с конкретной задачей: нужно запустить проект, собрать команду, оценить сроки, подготовить статус-отчёт, провести встречу, зафиксировать риск, согласовать изменение, принять результат или закрыть проект.

Приложения можно использовать как рабочий набор начинающего РП. Там собраны структуры документов, чек-листы, реестры и рабочие листы, к которым я буду возвращаться по ходу книги. Не пытайтесь внедрить всё сразу и не превращайте проект в склад шаблонов. Берите инструменты точно: в маленьком проекте достаточно нескольких простых документов, в крупном — набор можно постепенно расши-

рять и собирать полноценную систему управления.

Главная задача этой книги — не просто объяснить термины проектного управления, а дать рабочий маршрут: как начать проект, навести порядок в требованиях, сроках, ресурсах, коммуникациях и рисках, провести команду через исполнение, довести результат до приёмки, а сам проект — до закрытия без лишнего хаоса. Поэтому почти каждая глава отвечает не только на вопрос «**Что это такое?**», но и на вопрос «**Что с этим делать руководителю проекта завтра утром?**».

Управление проектами — это способ довести идею до результата при ограниченных ресурсах: времени, деньгах, людях, технологиях, внимании команды и поддержке заинтересованных сторон. Руководитель проекта постоянно балансирует между объёмом работ, сроками, бюджетом, качеством и рисками.

Проектное управление нужно не только крупным корпорациям. Почти любое значимое изменение в компании превращается в проект: запуск цифрового продукта, внедрение внутренней системы, автоматизация процесса, выход на новый рынок, работа с подрядчиками, оптимизация затрат или перестройка бизнес-процесса.

Проектов становится больше, сроки становятся плотнее, команды — распределённые, технологии — сложнее, а терпение заказчиков — короче. Все хотят быстрее, дешевле, качественнее и желательно вчера. Красиво звучит, но без

управления это быстро превращается в хаос с табличкой «**Мы почти успели**».

По данным Project Management Institute, в 2025 году в мире насчитывается около 39,6 миллиона проектных специалистов. При этом к 2035 году мировой рынок может столкнуться с дефицитом до 29,8 миллиона квалифицированных проектных специалистов, а глобальный спрос на проектные кадры может вырасти на 64% с 2025 по 2035 год.

Статистика провалов проектов тоже показывает важную вещь: проекты редко ломаются из-за одной ошибки. Обычно их разрушает комбинация факторов: слабые требования, плохое планирование, неуправляемые изменения, нереалистичные ожидания, проблемы коммуникации, отсутствие спонсора, слабая работа с рисками и отсутствие человека, который видит проект целиком.

Одним из главных факторов успеха или неудачи проекта остаются требования: насколько они понятны, согласованы, проверяемы и одинаково понимаются заказчиком, руководителем проекта и командой. Но только на требованиях всё не заканчивается. Проект может пострадать из-за нехватки ресурсов, неверной оценки сроков, плохой коммуникации, незрелости команды, отсутствия приоритетов или страха вовремя сказать: «**У нас проблема**».

Ниже собрал основные факторы, которые сильно влияют на успешность проекта:

— вовлечение участников проекта и определение заинте-

ресованных лиц;

- квалифицированная команда;
- понятный владделец продукта или заказчик результата;
- поддержка спонсора и руководства;
- чётко описанные требования;
- понятные и достижимые цели;
- реалистичное планирование сроков, ресурсов и бюджета;
- управление рисками;
- зрелая коммуникация;
- готовность команды и заказчика к изменениям;
- реалистичные ожидания;
- подходящая методология управления.

Факторы, перечисленные выше, мы разберём дальше в книге.

Эта книга не заменяет профессиональные стандарты и методологии, но помогает понять, как применять проектное мышление в реальной работе. Стандарты дают язык и рамки, а практика каждый день проверяет руководителя проекта на способность принимать решения, договариваться, видеть риски, защищать сроки и не терять управляемость в хаосе.

Профессиональное управление проектами помогает снижать риски и затраты, улучшать качество результата, повышать прозрачность работы и вовремя принимать решения. Без планирования, контроля, коммуникации и ответствен-

ного руководителя проект легко превращается в бесконечный поток правок, ожиданий, споров и переделок.

Руководитель проекта — это не просто человек, который ведёт календарь и спрашивает: «**Когда будет готово?**». Это специалист, который помогает бизнесу превращать идеи в результат, связывает заказчика, команду и заинтересованные стороны, управляет ограничениями и удерживает проект в рабочем состоянии.

Дальше мы начнём с базовых понятий: что такое проект, чем он отличается от операционной деятельности, какую роль играет руководитель проекта, какие бывают команды, из каких этапов состоит жизненный цикл проекта и как выбрать подход к управлению.

ГЛАВА 1. УПРАВЛЕНИЕ ПРОЕКТАМИ

Что вы узнаете: что такое проект и чем он отличается от операционной деятельности; зачем компаниям нужно управление проектами; какие бывают проектные команды; какую роль играет руководитель проекта; из каких этапов состоит жизненный цикл проекта; как выбрать подход к управлению — классический, гибкий или гибридный.

1.1. ЧТО ТАКОЕ ПРОЕКТ?

Начинающему РП важно быстро понять простую вещь: проект — это не любая задача и не любое дело в календаре. Проект начинается там, где нужно создать изменение: продукт, услугу, систему, процесс, внедрение или другой результат, которого раньше не было в нужном виде.

Согласно официальной терминологии:

— Первое определение утверждает, что проект — предприятие с определёнными датами начала и завершения, предпринятое для создания продукта или услуги (сервиса) в соответствии с заданными ресурсами и требованиями (ISO/IEC/IEEE 15288:2008 Systems and software engineering — System life cycle processes).

— Ещё одно определение данного понятия из свода знаний PMBOK утверждает, что проект (в управленческой деятельности) — временное предприятие, направленное на создание уникального продукта, услуги или результата.

Оба определения верны, но начинающему руководителю проекта важно не застрять в формулировках. В реальной работе проект отличается от операционной деятельности не красивым определением, а управленческими признаками: у него есть цель, начало, окончание, ограничения, ответственные люди и ожидаемый результат.

Операционная деятельность повторяется и поддержива-

ет стабильную работу бизнеса. Проект же создаёт изменение: новый продукт, сервис, систему, процесс, внедрение или другой результат, которого раньше не было в таком виде.

Цели проекта и операционной деятельности отличаются. Проект не должен длиться вечно: он заканчивается, когда цель достигнута или когда принято решение его остановить. Операционная деятельность, наоборот, продолжается постоянно — она поддерживает обычную работу бизнеса и повторяющиеся процессы.

Проще говоря, проект нужен там, где обычной операционной работы уже недостаточно. Когда компании нужно что-то изменить, создать или внедрить: запустить продукт, внедрить систему, перестроить процесс, сделать интеграцию или выполнить важную часть стратегического плана.

В компании проекты существуют, чтобы найти решение конкретной проблемы или задачи. В разработке ПО это может быть создание нового продукта, внутреннего сервиса, интеграции, доработка существующей системы или запуск решения для внешнего заказчика.

Проект начинают использовать, когда организации не хватает системного подхода для решения конкретной проблемы/задачи. Это разовое мероприятие, вероятно, не будет повторно использовано в будущем, что делает его одноразовым событием, следовательно, проектом. Проект соответствует следующим характеристикам:

— Проект имеет чёткие функциональные рамки.

— Проект может потребовать использования рискованных решений.

— Для реализации проекта могут потребоваться специальные инструменты, технологии, опыт или ресурсы, помимо того, что организация уже имеет.

— Для проекта может потребоваться уникальное расположение места реализации.

— Проект имеет ограниченные временные рамки. После завершения проект не продолжается в прежнем виде.

— Проект имеет отличительную особенность — последовательная разработка (Progressive Elaboration). **Последовательная разработка** — непрерывное улучшение и детализация плана по мере получения более подробной информации и более точных оценок в процессе исполнения проекта. Благодаря этому разрабатываются более точные и более полные планы, являющиеся результатом многократного повторения процесса планирования.

1.2. ПОЧЕМУ УПРАВЛЕНИЕ ПРОЕКТАМИ?

Некоторые компании начинают ценить управление проектами только после первого серьёзного провала. Пока всё более-менее едет, планы, риски, роли, протоколы и контроль изменений могут казаться лишней бюрократией. Но когда сроки сорваны, бюджет потрачен, заказчик недоволен, а команда не понимает, кто и что обещал, проектное управление внезапно перестаёт быть скучной формальностью. Современным компаниям приходится быстро реагировать на изменения и проводить внутренние преобразования. А любое серьёзное изменение лучше вести как проект: с целью, сроками, ответственными, ресурсами и контролем результата. Иначе оно легко превращается в неэффективное расходование времени, денег и сил.

Компании всё чаще вынуждены быстро разрабатывать и выводить на рынок новые продукты или услуги. Иначе растёт риск неэффективной работы, потери темпа и ослабления конкурентных позиций.

Управление проектами — это область деятельности, в ходе которой определяются и достигаются чёткие цели проекта, балансируются объём работ, ресурсы, время, качество и риски, а также выполняются планирование, организация, подбор команды и контроль выполнения работ. Это широко

распространённый подход, который могут использовать компании разного размера. Он опирается на работу качественно подобранной проектной команды, где каждый участник обладает навыками и опытом, необходимыми для движения проекта от инициации до завершения.

Далее собрал в список основные причины распространения проектного управления:

1. Сокращение жизненного цикла товаров.
2. Усиление конкуренции.
3. Рост неопределённости внешней среды.
4. Стремительное развитие технологий.
5. Возросшие требования потребителей.
6. Сокращение расходов.

Проектная команда нужна, чтобы проект не жил сам по себе и не расползался между отделами. У бизнеса остаётся фокус на управлении компанией, а у команды появляется понятная ответственность: собрать нужных людей, выполнить работы, оперативно решать проблемы по ходу проекта и довести результат до готовности.

1.3. ПРОЕКТНАЯ КОМАНДА И ЕЁ ВАРИАНТЫ

Как мы помним, проект — это деятельность. Значит, если упростить, управление проектом — это управление теми, кто эту деятельность выполняет.

Людей, которые будут выполнять эту работу, нужно собрать в проектную команду. Без команды замысел так и останется замыслом: кто-то должен взять задачи, договорённости и ответственность за результат. Учитывая написанное ранее, сформулирую, что такое проектная команда.

Проектная команда — это группа специалистов, собираемая на определённый период времени для выполнения проекта. Она создаётся для достижения поставленных целей и задач конкретного проекта и подчиняется проектному менеджеру.

Подбор и налаживание работы команды — это сложный и дорогостоящий процесс, поскольку затрачиваются деньги и время разных специалистов (в зависимости от структуры) на поиск нужных сотрудников, выстраивание взаимоотношений между ними и управление их деятельностью.

Далее приведу наиболее распространённые варианты проектной команды (без учёта конкретных специалистов внутри команд):

— **Вариант 1.** Проектная команда только из штатных

(inhouse) сотрудников;

— **Вариант 2.** Проектная команда из штатных сотрудников, включая распределённых по разным офисам и/или удалённый штат (работающие из дома);

— **Вариант 3.** Проектная команда из штатных сотрудников + аутстафф-команда;

— **Вариант 4.** Проектная команда из штатных сотрудников + одна или несколько команд на аутсорсе: им передают часть работ, а приёмку выполняют внутренние специалисты.

Варианты 1 и 2 понятные и пояснять их не буду. А вот варианты 3 и 4 более интересные: когда есть потребность, вы подключаете дополнительные ресурсы, а когда потребность исчезает — не держите их постоянно в штате.

Вариант 3 подразумевает, что выделенные специалисты управляют почти так же, как внутренние ресурсы, но подключаются только на нужный период. Вариант 4 подходит, когда часть работ можно передать внешней команде, заранее договорившись о результате, сроках, качестве и правилах приёмки.

Какой вариант выбрать — решать вам, исходя из потребностей компании, ограничений и удобства работы.

Далее хочу рассказать вам немного о составе команд, которые я чаще всего встречал в работе и которые являются самыми популярными. Начнём с того, что роли в команде могут существенно различаться. Состав команды будет зави-

сеть от многих факторов: размера компании, типа проекта, а также от того, создаёте вы что-то с нуля или дорабатываете существующее решение.

Как вы понимаете, факторов, а значит и комбинаций очень много, но я хочу рассказать вам о наиболее распространённых из них, которые используются в той или иной степени практически на любом проекте:

1. Состав «Упрощённый»

- Руководитель проекта.
- Программисты (Frontend/Backend или Fullstack/Mobile).
- Дизайнер.
- Аккаунт-менеджер.

2. Состав «Классический»

- Руководитель проекта.
- Технический лидер.
- Программисты (Frontend/Backend/Fullstack/Mobile).
- Дизайнер.
- Аккаунт-менеджер.
- Аналитик.

3. Состав «Классический +»

- Руководитель проекта.
- Технический лидер (иногда добавляют системного архитектора или один совмещает две роли).

- Программисты (Frontend/Backend/Fullstack/Mobile).
- Дизайнер.
- Специалист по продажам.
- Аналитик.
- Специалист внедрения и сопровождения.
- Поддержка.

4. Состав «Расширенный»

- Руководитель отдела разработки или Технический директор.
- Руководитель проекта (часто в таком составе РП находится в подчинении руководителя отдела или СТО).
- Системный архитектор.
- Тимлид/-ы.
- Программисты (Frontend/Backend/Fullstack/Mobile).
- Дизайнер/-ы и/или Дизайнеры UI/UX.
- Специалист/-ы по продажам.
- Аналитик/-и.
- DevOps-инженер/-ы (не всегда).
- Контент-менеджер/-ы.
- Специалист/-ы внедрения и сопровождения.
- Поддержка.

Замечу:

В небольших проектах или компаниях часто приходится объединять те или иные обязанности, исполняемые одной ролью, но указанные выше

составы послужат опорой при формировании вашей команды без долгих размышлений, правильно ли вы собрали команду или нет.

1.4. РУКОВОДИТЕЛЬ ПРОЕКТА. КТО ТАКОЙ? КАКИЕ У НЕГО ОБЯЗАННОСТИ?

Как вы видите, руководители проектов нужны во всех составах и являются важным звеном. Они выступают связующим звеном между пользователями/заказчиками и командой: помогают донести желания и потребности до аналитиков и других специалистов, чтобы затем превратить их в конкретные требования и техническое задание для всей команды. После передачи результата руководитель проекта также следит, чтобы договорённости по поддержке и доработкам были понятны и зафиксированы.

Руководителю проекта важно разбираться не только в сроках и задачах, но и в рисках, планировании, коммуникациях, ресурсах и качестве. Проектное управление опирается на сильную команду, но не должно превращаться в обучение сотрудников ценой провала проекта.

Руководитель проекта должен понимать способности и опыт каждого участника команды, чтобы правильно распределить обязанности и нагрузку. В хорошей проектной команде не должно быть людей, которые простаивают, работают впустую или постоянно перегружены.

Задача руководителя проекта — собрать сильные сторо-

ны участников в работающую систему. Он не обязан быть лучшим специалистом во всём. Но он обязан понимать, кто что умеет, кому какую работу можно доверить и как использовать ресурсы команды без хаоса и перегруза.

Хороший руководитель проекта может заметно повысить отдачу от вложенных ресурсов. Не магией, а управляемостью: убрать хаос в задачах, расставить приоритеты, не распылять команду, вовремя остановить лишние работы и сфокусировать людей на результате.

Иногда для этого нужно уменьшить число задач в работе, отказаться от второстепенного функционала или договориться с заказчиком о запуске первой версии продукта с ограниченным, но рабочим набором возможностей. Это лучше, чем бесконечно расширять объём, перегружать команду и так и не дойти до результата.

Например, я сталкивался со случаем, когда в отделе разработки ПО одной компании работало около 50 человек, но проекты всё равно проваливались по срокам минимум на 50%. Людей было много, занятость была высокая, совещаний и задач хватало всем. Со стороны могло казаться, что компания активно работает. Но результат говорил другое: качество оставалось низким, клиенты были недовольны, а сроки продолжали разъезжаться.

В результате разбора стало понятно, что проблема была не только в людях и не только в сложности проектов. Отсутствовали проектное планирование работ, подбор команды

под конкретный проект, нормальный контроль загрузки специалистов и управление потоком задач от заказчиков. Проще говоря, люди работали много, но сама система работы была неуправляемой.

После введения проектного управления проекты удалось вернуть в прогнозируемые сроки, повысить эффективность работы специалистов, уменьшить состав отдельных проектных команд и перераспределить людей между другими проектами. Это улучшило не только управляемость, но и финансовые показатели.

Главный вывод простой: если в проекте хаос, добавление людей не всегда помогает. Иногда оно просто делает хаос дороже.

В этом и ценность проектного управления: оно не делает проект лёгким, но делает его управляемым. У команды появляется понятная последовательность действий, у руководителя проекта — инструменты контроля, а у заказчика и руководства — прозрачность по срокам, рискам, результату и проблемам. Все эти меры дадут следующие результаты:

1. Снижение стоимости и сокращение сроков выполнения проекта.
2. Повышение качества результата и эффективности работы команды.
3. Повышение прозрачности процесса от идеи до приёмки.
4. Минимизация отклонения по срокам.

5. Повышение удовлетворённости владельцев, инвесторов, повышение мотивации персонала.
6. Усиление конкурентной позиции компании.

Вы должны понимать: проектное управление — не волшебная таблетка, после которой сразу станет хорошо. Это образ жизни руководителя проекта и всей компании. Ему можно и нужно учиться. И чем раньше компания перестаёт управлять проектами «на ощущениях», тем меньше денег, времени и нервов она сжигает впустую.

1.5. ЧЕТЫРЕ ЭТАПА УПРАВЛЕНИЯ ПРОЕКТОМ

Жизненный цикл проекта — это этапы, которые проходит проект с момента создания до завершения. Вы будете выполнять поэтапные действия, превращая идею в готовый к сдаче результат, который можно принять, передать и использовать.

Вообще, в жизненном цикле проекта выделяют пять этапов управления проектом:

- инициация;
- планирование;
- исполнение (выполнение);
- мониторинг;
- закрытие (завершение).

Вы явно зададите вопросы — **«Как пять?!»**, **«Разве в названии не четыре указано?!»**. И будете правы, но ошибки тут нет, я рассматриваю исполнение и мониторинг, как неразрывно связанные процессы одного этапа — **«Исполнение»**. Это взаимодействие я изобразил на рисунке 1. Приведу пример, поясняющий суть мысли. Невозможно приготовить сложное блюдо, всего лишь один раз взглянув на рецепт. Вы постоянно будете заглядывать в него и сверяться, в какой последовательности и какие ингредиенты до-

бавлять, сколько готовить, какая температура должна быть и так далее.

Вы будете внимательно следить за приготовлением, чтобы получить блюдо, описанное в рецепте. В этом примере приготовление можно считать проектом, блюдо — результатом, а контроль времени, объёма ингредиентов и последовательности действий — частью этапа «Мониторинг», который невозможно отделить от этапа «Исполнение». Работа над любым другим проектом идёт примерно так же. Конечно, это упрощённый пример, в проектной деятельности руководителю проекта придётся контролировать как минимум:

- Соблюдение сроков.

- Чтобы команда делала только то, что положено, не хватаясь за лишние работы.

- Чтобы расходы оставались в рамках бюджета.

- Все действия ведут к изначальной цели (особенно в больших или динамично меняющихся проектах легко можно потерять из виду важные моменты или вообще общую картину).

Уточню:

Не стесняйтесь использовать специализированное ПО для управления проектами. С ним мониторинг станет проще, так как процессы будут нагляднее, а вся информация и обсуждения, связанные с проектом, будут храниться в одном месте и станут доступны для всей команды.

Понимание жизненного цикла управления проектами позволит команде действовать не наугад, хватаясь за первые попавшиеся задачи, а работать над проектом стратегически и организованно, отслеживая ход работы. Ведение проекта по точным планам позволит быстрее завершить его, сведя к минимуму непредвиденные препятствия.

Сложив все эти преимущества, вы получите главное — команда сможет быстрее выполнить успешный проект и не выйти за рамки бюджета.



Рисунок 1 — Этапы управления проектом

Каждый этап должен быть доведён до понятного резуль-

тата, прежде чем проект перейдёт дальше. Нельзя нормально планировать то, что не прошло инициацию, и нельзя качественно закрыть проект, который не был доведён до приёмки. Далее коротко разберём каждый этап.

1.5.1. ЭТАП «ИНИЦИАЦИЯ»

Этап инициации обозначает начало нового проекта. На этом этапе организация определяет цели, границы проекта, ожидаемые результаты, средства и технологии достижения целей, предварительные затраты и команду. Часто на данном этапе (в конце) определяют всех заинтересованных лиц и руководителя проекта. Нужно чётко понимать, что эти данные приблизительные и чаще всего включают максимальный вариант оценки по всем пунктам.

Вы, может быть, уже слышали или только узнаете такое понятие, как ballpark. Это приблизительная оценка, смета с описанием того, что будет реализовано, сколько времени потребуется на это и сколько будет стоить, какие типы специалистов будут задействованы и какие сроки. Как минимум такой документ должен быть на выходе данного этапа.

По-хорошему, основным результатом этого этапа является более подробный документ — устав проекта, который содержит:

- Требования, удовлетворяющие потребности, пожелания и ожидания заказчика, спонсора и других заинтересованных сторон проекта.
- Общее описание проекта или требований к продукту, который является результатом выполнения проекта.
- Цели (по сути, это обоснование проекта).

- Сведения о руководителе проекта, проектной команде (включая роли и обязанности).
- Данные по клиенту, представители клиента и контактные данные.
- Основные этапы проекта.
- Риски и возможные проблемы.
- Список заинтересованных сторон и их участие в проекте.
- Функциональные подразделения организации и их участие в проекте (если есть деление).
- Ограничения, накладываемые организацией, окружением и внешней средой.
- Обоснование проекта, включая бизнес-обоснование и данные о ROI (Return on Investment).
- Суммарный бюджет проекта.

Готовую структуру такого документа можно посмотреть в приложении А.1. «Устав проекта: структура». Она поможет не начинать устав с пустого листа и не забыть базовые разделы: цели, границы, роли, риски, ограничения и ключевые параметры проекта.

Чётких требований к данному документу нет. В зависимости от компании и проекта список пунктов может меняться. В приведённом списке выше указано всё нужное и важное, устав проекта разрабатывается итерационно и может иметь несколько редакций, постепенно уточняющих, добав-

ляющих или удаляющих пункты.

Данный этап является важнейшим, так как на нём собираются все данные по будущему проекту. Молодые специалисты часто не уделяют ему достаточно внимания и не прорабатывают. Бывает, что и люди с большим опытом часто не делают качественную проработку устава.

Заключительным шагом начального этапа является заполнение формы обзора проекта. Она позволит руководителю проекта эффективно общаться с заказчиками проекта, которые хотят получать регулярные обновления о ходе проекта. Обзоры проектов, как правило, подготавливаются в конце этапов проекта. Здесь не говорим про показ функционала, реализуемого в ходе проекта.

На шаге инициации проекта форма обзора проекта будет включать обновления информации о том, запланировано ли завершение проекта и какие сроки, есть ли у организации достаточно ресурсов, а также какие-либо нерешённые вопросы, которые на данный момент не позволяют запустить проект в работу.

Основная цель формы обзора проекта — получить одобрение заказчика, чтобы перейти к следующему этапу.

1.5.2. ЭТАП «ПЛАНИРОВАНИЕ»

Основная цель этого этапа — составить детальный **план действий**.

На нём выделенная на проект команда (да, вы правильно поняли — до этого этапа команда уже должна быть собрана; или как минимум основная часть команды) и руководитель проекта начинают прорабатывать материалы более подробно. План действий представляет собой письменный документ, как и бизнес-план, — это «живой» документ, который будет изменяться по мере необходимости в ходе реализации проекта.

К этому плану стоит относиться как к дорожной карте, а не как к красивому файлу, который один раз согласовали, оформили и успешно забыли в папке «Проекты».

План показывает масштаб проекта, основные этапы, задачи, ответственных и предварительные сроки. В начале он может быть укрупнённым: несколько функциональных блоков, примерные оценки, понятные зависимости. Но к концу планирования он должен стать рабочим планом-графиком, по которому можно управлять проектом.

Задача руководителя проекта — не просто один раз создать план, а регулярно сравнивать его с фактическим состоянием задач. Для этого удобно использовать систему управления задачами: Jira, Redmine или другой инструмент, при-

нятый в компании.

Основной план будет включать в себя несколько подпланов, отражающих конкретные аспекты проекта:

— **План ресурсов.** Это документ, содержащий важную информацию обо всех ресурсах, которые понадобятся команде, от затрат на рабочую силу до инструментов, чтобы сделать проект успешным. Одним из наиболее важных аспектов плана ресурсов является расписание ресурсов. Этот график позволяет проектной команде планировать, сколько ресурсов понадобится и для каких частей проекта. В плане ресурсов перечислены типы рабочей силы, необходимые проекту, число ролей, которые необходимо заполнить, и обязанности каждой роли. Он должен также перечислить количество оборудования и других материалов, необходимых проекту для продвижения вперёд. Но это не всё. Настоятельно рекомендую учесть в этом плане как минимум:

- праздники,
- отпуска,
- возможные отгулы,
- больничные (вы работаете с людьми, знаете их и сможете спрогнозировать, как часто в месяц они отсутствуют),
- риски (например, задержки у клиента в предоставлении обратной связи),
- занятость на проектах (задействованы ли на других проектах и в каком объёме).

Этот список можно продолжать, но для старта достаточ-

но и этих пунктов. Может возникнуть вопрос: «**Зачем так много сложностей?**» На самом деле это не сложности, а нормальная подготовка. Чем лучше вы понимаете ресурсы на старте, тем меньше неожиданностей получите в ходе проекта. Перечислю, что зависит от ресурсного плана:

- прибыль компании от проекта,
- успех проекта (люди работают над проектом),
- своевременный найм сотрудников (а это работа HR),
- подготовка рабочего места (а это работа поддержки),
- эффективность работы компании.

— **Финансовый план.** Этот план основывается на ресурсном плане и служит для детализации суммы денежных средств, которая потребуется для реализации проекта. Основная цель финансового плана — обеспечить основу для сравнения общего бюджета проекта и его фактической стоимости (сколько потратили на реализацию). Давайте рассмотрим основные пункты финансового плана:

— **Административные расходы** — оформление юридического лица и налоги, аренда помещения. Рассчитайте, какую сумму вам нужно будет зарезервировать на все бюрократические процедуры и уплату налогов. Этот пункт не всегда обязателен: возможно, вы только запускаете дело, а компании или помещения ещё нет.

— **Зарплата сотрудников.** Нужно учесть зарплаты всех участников проекта с учётом тех, кого вы ещё планируете нанять или тех, кого вы планируете привлекать разово.

— **Расходы на закупку материалов или оборудования.** На начальном этапе постарайтесь быть аккуратным: иногда сложно спрогнозировать, сколько и чего может потребоваться. Зарезервируйте денежные средства с запасом, но покупайте только необходимое. Пока нет утверждённого ТЗ, стоит уточнять, нужно ли покупать какой-то софт, библиотеки или специфичное оборудование, потому что суммы могут быть большими, и это может сильно изменить бюджет.

— **Транспортные расходы.** В этом пункте учтите стоимость доставки материалов к вам, выезды ваших специалистов к клиенту (а это может быть дорого и не один раз, если клиент в другом городе). Определите, с какими службами доставки вы будете работать, включите ли вы стоимость доставки в стоимость проекта или заказчик будет оплачивать её отдельно. В моей практике были случаи доставки курьерскими службами документов, образцов техники (терминалов), образцов продукции — от пищевой до бытовых средств. Поверьте, это было дорого.

— **Бюджет на продвижение проекта.** В этом пункте обязательно учтите бюджет на разработку фирменного стиля, сайта, упаковку товара, работу фотографа, рекламу в социальных сетях и участие в маркетах (может оказаться, что нужно покупать аккаунты в магазинах приложений).

Обязательно обсудите с заказчиком ожидания по дизайну, фирменному стилю, упаковке, сайту, фотографиям, рекламе и другим смежным работам. Часто клиент считает, что

«дизайн продукта» уже включает всё: фирменный стиль, материалы для продвижения, фотографии, тексты и оформление. На практике это могут быть отдельные работы, отдельные сроки и отдельный бюджет.

Такие расходы лучше проговорить на старте и заложить хотя бы 5—10% на непредвиденные затраты. Дизайн, фирменный стиль, подбор фотографий и рекламные материалы редко согласовываются с первого раза. И лучше честно учесть это в плане, чем потом делать вид, что никто не ожидал задержек.

— **План качества.** План позволяет группе оценить качество конкретных достижений в рамках проекта и определить, соответствует ли требованиям выпущенный продукт. Руководитель проекта может использовать план качества, чтобы установить в нём сроки релизов версий для контроля качества и деятельности всей команды, чтобы удержать проект в нужном русле. Руководитель проекта в этом документе продублирует сроки сдачи проекта (финальные) и отметки о смещении сроков и причинах.

— **План рисков.** Данный план содержит все риски, связанные с проектом. Разбейте их по категориям, а затем по приоритетам, так будет удобнее. Это поможет команде в определении того, какие риски наиболее вероятны и важны, позволив более эффективно распределить время или принять решение об устранении проблемы. В этом плане стоит описать, как каждый риск повлияет на проект, если его не

снизить или не устранить. Руководитель проекта и команда используют план рисков, чтобы заранее продумать превентивные меры, подготовить действия на случай проблем и отслеживать риски на протяжении всего проекта. План рисков является важным шагом в процессе планирования, так как он может значительно снизить вероятность неудач.

— **План приёмки** служит мостом между результатами работы проектной команды и заказчиком. В плане будет отображено состояние разрабатываемого проекта. Он также нужен, чтобы помочь определить любые ресурсы, которые могут понадобиться для тестирования. В конце концов, это официальное соглашение между заказчиком и исполнителем, в котором будут указаны детали всех результатов проекта и определены стандарты, необходимые для приёмки заказчиком. Исполнитель может использовать этот план, чтобы определить, как он будет тестировать выходные данные и функционал проекта.

К плану приёмки полезно подготовить «**Сопроводительное письмо**». Используйте его, когда передаёте заказчику промежуточную сборку, макет, документ, модуль или другой результат для проверки.

В сопроводительном письме коротко укажите, что именно передаётся, что нужно проверить и где заказчик может оставить комментарий: принято / не принято / нужны исправления. Если результат важный, лучше получить письменное подтверждение. Мы все люди: договорённости забываются,

письма теряются, а согласованный документ помогает спокойно вернуться к фактам.

Для маленького проекта иногда достаточно письма по почте. Для крупного проекта или финальной сдачи лучше использовать отдельный файл: с перечнем передаваемых материалов, статусом проверки, замечаниями и итоговым решением заказчика.

— **План коммуникации.** Его цель — помочь руководителю проекта и команде передавать важную информацию нужным людям, в нужное время и правильным способом. На этапе планирования достаточно зафиксировать базовые правила: кто участвует в коммуникациях, какую информацию получает, как часто, через какие каналы и кто отвечает за обновление договорённостей.

Подробно коммуникационный план, работу со сложными стейкхолдерами, типовые ошибки общения, обратную связь и особенности управленческой коммуникации мы разберём в отдельной главе «Коммуникация. 90% работы РП».

— **План закупок** помогает заранее определить, какие внешние ресурсы понадобятся проекту: оборудование, лицензии, материалы, услуги, подрядчики или экспертиза. В нём фиксируют, что нужно получить, когда это понадобится, кто согласует закупку, кто отвечает за взаимодействие с поставщиками и какие риски возникнут при задержке. Этот план важно согласовать с заказчиком, чтобы потом не было споров в стиле: «Мы думали, вы это купите и уже вклю-

чили в стоимость проекта».

Уточнение:

Все эти планы разделены для удобства восприятия и работы с ними. Но это не значит, что только так нужно разделять. В моей практике были случаи, где это было разбито на пару документов.

— **Тендерный процесс.** После завершения плана закупок следующим шагом становится подключение поставщиков услуг и/или оборудования. Команды управления проектами обычно называют это тендерным процессом. Он включает в себя разработку нескольких документов, которые помогут команде управлять закупками и выбрать лучших поставщиков:

— ТЗ определяет и уточняет тип работ, тип поставщика, ресурсы (необходимые команде), график сроков и копию условий оплаты.

— Далее идёт документ запроса информации (RFI). Этот документ помогает команде определить потребности.

— После запроса информации запрашивают документ с предложениями (RFP), он поможет команде выбрать лучшего поставщика, опираясь на конкретные предложения.

— Далее идёт контракт с поставщиком, который по существу является документом, в котором излагаются условия между организацией и поставщиком. В контракте должны быть конкретно указаны все ресурсы, которые будет предо-

ставлять поставщик, все сроки выполнения и информация о выставлении счетов. В нём также должны быть изложены условия договора для обеих сторон. Эти стороны затем и подпишут документ.

— Наконец, команде потребуются тендерные формы для отслеживания всего, что происходит в процессе выбора поставщиков.

Заключительным блоком работ этапа планирования является ещё один обзор проекта. В данном блоке работ в форме обзора проекта следует указать:

- планируется ли завершить проект в срок;
- соответствует ли он бюджету на сегодняшний день;
- устранены или нет какие-либо существующие препятствия и риски.

В нём также необходимо перечислить любые ресурсы и материалы, использование которых было утверждено в процессе планирования.

Подробно планирование разберём в следующей главе.

1.5.3. ЭТАП «ИСПОЛНЕНИЕ»

Этап выполнения работ, или этап исполнения, — самая практическая часть проекта. На этом этапе проектная команда приступает к созданию конечного результата: продукта, сервиса, системы, внедрения или другого результата, ради которого проект был запущен.

Именно здесь всё планирование начинает проверяться реальностью. Становится видно, насколько качественно были собраны требования, правильно ли оценены сроки, достаточно ли ресурсов, реалистичен ли бюджет и готова ли команда работать в выбранном темпе.

Важно понимать: исполнение проекта — это не просто **«делать задачи по плану»**. Это постоянное управление отклонениями. В реальном проекте почти всегда появляются изменения, уточнения, риски, задержки, вопросы по качеству, проблемы коммуникации, зависимость от поставщиков и необходимость принимать управленческие решения в условиях давления.

На этапе исполнения руководитель проекта управляет несколькими направлениями одновременно:

- сроками и графиком работ;
- затратами и фактическими расходами;
- качеством результата;
- изменениями требований и объёма работ;

- рисками;
- поставщиками и подрядчиками;
- коммуникациями;
- приёмкой промежуточных результатов;
- состоянием и мотивацией команды.

Финальной частью этапа исполнения является обзор проекта и заполнение формы обзора. В ней руководитель проекта фиксирует, как проект продвигается к завершению, есть ли проблемы на текущий момент, какие препятствия или риски были устранены и какие действия были официально запланированы.

В этой обзорной главе важно запомнить главное: этап исполнения показывает, насколько проект действительно управляем. Подробно о том, как вести проект, когда всё пошло не по плану, как управлять изменениями, контролировать график, защищать качество, работать с командой «в бою» и использовать инструменты руководителя проекта, мы разберём в отдельной главе «Исполнение: как вести проект, когда всё пошло не по плану».

1.5.4. ЭТАП «ЗАКРЫТИЕ»

Мы подошли к завершающей стадии процесса управления проектом, заключительному этапу.

Этот этап позволяет руководителю проекта и команде официально закрыть проект, передать итоговую информацию руководству и зафиксировать результат работы.

На этом этапе проектная команда также передаёт всю проектную документацию клиенту архивом, сам проект закрывается, отношения с поставщиками прекращаются, проектная команда расформировывается. Тут стоит уточнить — написанное выше не ведёт к увольнению сотрудников (конечно, если увольнение не планировалось), это просто освобождение человеческих ресурсов, чтобы появилась возможность сформировать проектную команду под новый проект или перераспределить специалистов по текущим (усиление текущих команд). Не забывайте, что после сдачи проект нужно поддерживать, и на это также необходимо зарезервировать время специалистов. Если клиент оплатил поддержку, то при роспуске команды нужно оставить ряд специалистов на проекте для осуществления работ по поддержке или выполнения мелких доработок в рамках договорённостей.

Также нужно провести **ретроспективу проекта** — финальный разбор того, что получилось, что не получилось и какие выводы нужно забрать в будущие проекты. Это не

встреча ради разговора, а способ превратить опыт команды в управленческие улучшения.

Ретроспективы полезно проводить не только в самом конце, но и после важных этапов, спринтов или релизов. Финальная ретроспектива даёт команде ощущение завершенности и помогает зафиксировать опыт, ошибки, удачные решения и практики, которые стоит повторить.

Скажу больше: хотя основной этап планирования завершается в начале проекта, планы всё равно будут уточняться и корректироваться на протяжении всего жизненного цикла. Некоторые элементы будут возвращаться на разных этапах, но каждый раз — с другой задачей и смыслом.

Подробно этап закрытия разберём в главе 9.

1.6. ВЫБОР МЕТОДОЛОГИИ И РАБОТА С ПОДХОДОМ К ПРОЕКТУ

1.6.1. МИР МЕТОДОЛОГИЙ: КЛАССИКА (WATERFALL) VS ГИБКОСТЬ (AGILE/SCRUM)

Что вы узнаете: какие основные подходы к управлению проектами существуют и как выбрать подходящий вариант для вашего проекта.

Выбор методологии — это первый реальный тест на зрелость руководителя проекта. От того, насколько точно вы сможете согласовать подход с командой, заказчиком и ключевыми заинтересованными лицами, зависит успешность всего проекта.

До этого мы говорили об этапах проекта: инициации, планировании, исполнении и закрытии. Такое движение по последовательным шагам ближе к классическому подходу, который часто называют каскадным, или Waterfall. Он хорошо работает там, где требования понятны и стабильны. Например, при строительстве дома по готовому проекту или при внедрении заранее описанного регламента.

Но на практике не менее важно не только понимать этапы,

но и выбрать, как именно вы будете проходить эти этапы. Что делать, если заказчик сам не до конца понимает, какой продукт ему нужен? Что делать, если рынок быстро меняется, а требования уточняются уже по ходу работы? В таких ситуациях помогают гибкие подходы, или Agile. Самая известная из гибких методологий — Scrum.

Новичку важно запомнить простую мысль: классика хороша для стабильных проектов, гибкость — для неопределённых. Выбранный подход влияет на планирование, контроль, взаимодействие с командой, работу с заказчиком, управление рисками и частоту обратной связи.

Ключевые отличия Waterfall и Scrum для новичка представлены в таблице 1.

Аспект	Классика (Waterfall)	Scrum (гибкая методология)
Подход	Строгая последовательность этапов. Следующий этап начинается после окончания предыдущего.	Работа идёт короткими циклами — спринтами по 1–4 недели. В конце каждого спринта команда показывает готовый кусок продукта.
Изменения	Дороги и сложны. После стадии планирования их стараются избегать или оформлять через формальный процесс изменений.	Ожидаемы и встроены в процесс. В конце каждого спринта план может корректироваться на основе обратной связи.
Фокус	Следование изначальному плану, срокам, бюджету и документации.	Быстрая реакция на обратную связь и регулярное получение ценного результата для заказчика.
Роль руководителя проекта	Руководитель проекта планирует, контролирует, управляет сроками, бюджетом, рисками и отчётностью.	Скрам-мастер выступает фасилитатором: помогает команде работать, убирает препятствия, но не управляет командой напрямую в классическом смысле.

Таблица 1 — Ключевые отличия Waterfall и Scrum для новичка

Эта таблица помогает быстро сравнить ключевые аспекты подходов и понять, где роль руководителя проекта наиболее активна.

1.6.2. КАК ВЫБРАТЬ?

— Выбирайте классический подход, если требования понятны, проект не очень большой, заказчик не планирует активно участвовать в работе, а изменения должны проходить формально.

— Присмотритесь к Scrum, если в проекте много неопределённости, заказчик готов регулярно смотреть результат и давать обратную связь, а продукт нужно быстро вывести на рынок хотя бы с минимальным функционалом.

Запомните:

— Нет одной «правильной» методологии. Есть методология, подходящая под конкретный проект, команду, заказчика и ограничения бизнеса.

— Методология нужна не ради красивого названия и не ради модного слова в презентации, а ради управляемости проекта.

— В реальности часто используется гибридный подход: часть проекта управляется классически, а часть — гибко.

1.6.3. ГИБРИДНЫЕ ПОДХОДЫ: WATERFALL + AGILE

На практике чистый Waterfall или чистый Scrum встречаются редко. Чаще используется гибридный подход. Например:

- Инициацию и высокоуровневое планирование проводят по классическому подходу: фиксируют бюджет, сроки, ключевые требования и Устав проекта.

- Исполнение разбивают на Agile-спринты по 2—4 недели, внутри которых команда работает гибко и регулярно показывает инкремент продукта.

- Сдачу результата и закрытие проекта снова формализуют: подписывают документы, закрывают договорённости, фиксируют итоги и уроки.

Гибридный подход даёт лучшее из двух миров: предсказуемость для бизнеса и гибкость для команды. Руководителю проекта важно одинаково уверенно работать и с формальными документами, и с гибкими командами.

1.6.4. ИСТОРИИ УСПЕХА РЕАЛЬНЫХ ПРОЕКТОВ

Ниже — несколько кратких примеров проектов, в которых подход к управлению выбирали не «по моде», а под реальность задачи. Это не полный разбор кейсов, а иллюстрация главного принципа: методология должна помогать управлять проектом, а не украшать презентацию.

1. Проект: стартап по доставке еды (Agile/Scrum)

Компания имела сайт и решила развиваться дальше — создать мобильное приложение для заказа еды. На старте было много неопределённости: какие функции действительно нужны пользователям, как они будут оформлять заказ, что окажется удобным, а что останется красивой идеей на бумаге.

В такой ситуации опасно было надолго закрыться в разработке и через несколько месяцев показать рынку «идеальный» продукт, который никому не нужен. Поэтому команда выбрала Agile-подход: быстро собрать MVP, показать пользователям, получить обратную связь и постепенно дорабатывать продукт.

Быстрый запуск первой версии позволил проверить концепцию, убрать лишнее, усилить востребованные функции и двигаться не от фантазий команды, а от поведения реальных пользователей.

Ключевой фактор успеха: быстрый вывод MVP и регулярная обратная связь с рынком.

2. Проект: создание серверной на стороне клиента (Waterfall)

Компания, развивающая программное обеспечение для трансграничных платежей, решила создать серверную на своей территории и перенести туда ряд ключевых продуктов и часть внутренней инфраструктуры.

Здесь было мало места для импровизации. Требования к помещению, оборудованию, доступам, безопасности, срокам, поставщикам и ответственности нужно было определить заранее. Ошибка в планировании могла привести не к фразе «подумаем в следующем спринте», а к простою, лишним расходам и рискам для бизнеса.

Поэтому классический каскадный подход был уместен: заранее описать требования, согласовать план, проработать риски, синхронизировать подразделения и последовательно пройти этапы проекта.

Ключевой фактор успеха: чёткое планирование, контроль рисков и фиксированные требования.

3. Проект: оптимизация логистических маршрутов международной торговой сети (Scrum + Kanban)

В проекте по оптимизации логистических маршрутов было важно не только разработать решение, но и постоянно видеть поток работ: какие гипотезы проверяются, где возникают задержки, какие данные уточняются, какие маршруты

требуют пересмотра.

Scrum помог двигаться итерациями и регулярно проверять результат, а Kanban дал прозрачность по потоку задач и узким местам. Команда могла быстро видеть, что застряло, где нужна помощь и какие изменения стоит вносить в первую очередь.

Такой подход помог сократить издержки, улучшить качество обслуживания клиентов и быстрее адаптироваться к рыночным условиям.

Ключевой фактор успеха: итеративность, прозрачность и визуализация потока работ.

Главный вывод: методология — это инструмент управления, а не самоцель. Хороший руководитель проекта выбирает подход под реальность проекта, а не пытается любой проект загнать в любимую схему.

1.7. ЧТО ДЕЛАЕТ ПРОЕКТ УСПЕШНЫМ?

Выше мы уже несколько раз говорили об успешном проекте. Но что вообще значит «успех» в управлении проектами?

В широком смысле успешный проект — это проект, который дал качественный результат, был принят заказчиком и не развалился по срокам, бюджету, объёму и ожиданиям.

Качество и одобрение клиента всегда завязаны на три вещи: **объём, время и стоимость**. Руководитель проекта не может волшебным образом угадать точный срок и бюджет с первого разговора, особенно если у клиента пока только «хотелки». Но он может начать с главного — определить объём.

Объём показывает, за что проектная команда отвечает, а за что не отвечает. Пока объём не зафиксирован, сроки и стоимость будут гаданием. Сначала нужно понять, какой результат нужен заказчику, какие функциональные блоки входят в проект, какие ограничения есть и что точно не входит в работу. После этого проект можно разложить на задачи, оценить ресурсы, заложить риски, учесть тестирование, согласования, отпуска, занятость людей на других проектах и только потом говорить о реалистичных сроках.

Проще говоря: сначала границы, потом задачи, потом оценка, потом график. Не наоборот.

Но есть важный момент: сроки можно нормально оценивать только тогда, когда задачи действительно поставлены. Не названы, не брошены в чат, не сформулированы на уровне **«сделайте нормально»**, а именно **поставлены**.

Для руководителя проекта качество постановки задачи — это не мелочь и не бюрократия. Это основа управляемости. Плохо поставленная задача ломает оценку сроков, создаёт переделки, провоцирует конфликты и делает приёмку результата субъективной.

Поэтому перед передачей задачи в работу её стоит быстро проверить: достаточно ли она понятна, обеспечена ресурсами, связана с целью проекта, достижима и проверяема на выходе. Для такой проверки я использую собственный чек-лист — КРЕДО.

Авторский чек-лист КРЕДО

На практике проблема часто возникает не потому, что команда плохо работает, а потому, что задача изначально поставлена не как задача, а как пожелание, эмоция или общее направление мысли.

«Нужно улучшить отчёт», «Надо ускорить обработку», «Сделайте нормально», «Поправьте интеграцию», «Разберитесь с ошибкой» — всё это может звучать как задача, но управлять этим почти невозможно. Непонятно, что именно нужно сделать, кто принимает результат, какие есть ограничения, как измерить готовность и по каким критериям работа будет считаться выполненной.

Для проверки качества постановки задачи я использую авторский чек-лист КРЕДО. Я разработал его как практическую доработку логики SMART/SMARTER, но адаптировал не под постановку личных целей, а под проектные, продуктовые и ИТ-задачи.

SMART и SMARTER помогают думать о цели: конкретна ли она, измерима ли, достижима ли, релевантна ли и ограничена ли по времени. Но в проектной работе одной цели мало. Задача должна быть не только понятной, но и обеспеченной ресурсами, связанной с причиной, достижимой в текущих ограничениях и проверяемой на выходе.

КРЕДО помогает быстро проверить, можно ли такую задачу отдавать в работу:

К — конкретная. Понятно, что именно нужно сделать и в каком контексте.

Р — ресурсно обеспеченная. Понятно, кто исполнитель, какие сроки, какие доступы, данные, материалы или другие ресурсы нужны для выполнения.

Е — единая. Задача связана с целью проекта, запросом бизнеса, изменением, требованием или другой понятной причиной. Она не висит в воздухе.

Д — достижимая. Понятны ограничения, риски, зависимости, блокеры и примерная оценка трудозатрат.

О — оцениваемая. Понятно, по каким критериям будет проверяться результат и как команда поймёт, что задача действительно выполнена.

Плюс такого подхода простой: он снижает двусмысленность. Чем лучше поставлена задача, тем меньше переделок, споров, скрытых ожиданий и фраз вроде: «**Я думал, нужно было сделать не так**».

КРЕДО не является международным стандартом и не заменяет профессиональное мышление руководителя проекта. Это мой авторский рабочий инструмент, который помогает быстро проверить качество постановки задачи перед тем, как отдавать её в работу.

Полную рабочую версию инструмента я вынес в приложение А.12. «Чек-лист КРЕДО для постановки задачи». Используйте его при постановке задач команде, подрядчику, аналитику, разработчику или себе — особенно когда задача кажется «и так понятной».

Хочу вернуться к важному параметру, который упомянул выше — «Область применения», обычно она определяется на первом этапе. Наиболее важной целью области применения является то, что она устанавливает ограничения на функционал в проекте (а иногда и больше). Замечу, что всё, что находится за пределами области применения и допустимого объёма работ, может оказать негативное влияние на результат проекта.

Давайте опишу ещё одну проблему — появление новой и незапланированной работы в середине итерации или проекта. Подобное явление называют — «**ползучесть области**».

В гибких методологиях **ползучесть области** — действи-

тельно проблема: новые или незапланированные задачи добавляются прямо в середине итерации, вместо того чтобы сначала внести их в общий список задач проекта. Все гибкие методологии решают это посредством формальных процессов и церемоний. В методологии Скрам (Scrum), например:

1. Новая задача обычно должна вводиться только во время планирования Спринта (Sprint).

2. Новая задача, которая имеет приоритет над текущей задачей, требует раннего завершения текущего Спринта и возврата к планированию нового Спринта.

3. Новая задача в проекте должна быть приоритетной для Владельца Продукта в сотрудничестве с заинтересованными сторонами, так что ползучесть области на уровне проекта управляется договорённостями и соглашениями.

Немного поясню термины. В Скраме итерации называют «спринтами», они представляют собой фиксированные блоки работ (задач).

«Ползучесть области» в традиционном бизнес-смысле этого термина — расширение общего времени выполнения проекта за счёт добавления новых задач в проект, тем самым влияя на график, как правило, увеличивая его.

Можно смело сказать: лишние работы с высокой вероятностью ведут к провалу или, как минимум, к незапланированным расходам бюджета проекта. Но есть потребности бизнеса, которые игнорировать нельзя. В связи с этим предлагаю модификацию 1 и 2 пунктов из списка выше.

Если новая задача срочная и её нужно реализовать прямо сейчас, стоит оценить её и посмотреть, хватает ли запаса времени, чтобы добавить задачу в спринт и не сломать текущий план. Если запаса времени не хватает, нужно понять, какую задачу можно убрать и на её место поставить новую. Главное — чтобы продолжительность была сопоставимой и клиент согласовал замену. Более подробно Скрам (Scrum) стоит рассматривать отдельно.

Главное — не делать вид, что новые задачи сами аккуратно встроятся в проект. Так почти никогда не бывает. Любое изменение нужно честно оценить, обсудить и понять, что оно двигает: сроки, бюджет, объём работ или нагрузку команды. Тогда даже нестандартный подход будет работать лучше, чем формальная методология, которую все красиво называют, но никто по-настоящему не использует.

По поводу управления изменениями подробно поговорим в главе 7, а об успешности проекта — в главе 3.

1.8. РАБОТА БЕЗ УПРАВЛЕНИЯ ПРОЕКТАМИ. ПОУЧИТЕЛЬНАЯ ИСТОРИЯ

Давайте посмотрим, что происходит, когда у проекта вроде бы всё есть: договор, срок, бюджет, исполнитель и заказчик. Не хватает только нормального управления объёмом, требованиями и изменениями.

Нужно было разработать большую CRM-систему. По договору был зафиксирован год работ и фиксированный бюджет. Роль руководителя проекта отдали непрофильному специалисту. Требования заказчика почти напрямую превращались в задачи для разработки: без нормальной проверки, описания, декомпозиции, критериев приёмки и оценки влияния на сроки.

Разработчики реализовывали так, как понимали. Заказчик периодически смотрел сборки, выдавал новые пожелания и говорил: «**Нет, мы хотели не это**». Всё это ставилось в работу без оценки влияния на сроки, бюджет и объём.

Проблема стала очевидной только к концу срока. Бюджет закончился, заказчик отказывался принимать результат, команда была вынуждена переделывать уже сделанное, а стоимость проекта для исполнителя выросла примерно в три раза. И самое неприятное — пока команда спасала этот про-

ект, она не могла нормально работать над новыми.

Этот проект не провалился в один день. Он проваливался постепенно: каждый раз, когда новое пожелание превращалось в задачу без оценки; каждый раз, когда результат принимался «на глаз»; каждый раз, когда никто не задавал неудобный вопрос: «Что это меняет в сроках, бюджете и объёме?»

Этот проект мог закончиться иначе, если бы с самого начала были понятны область работ, требования, сроки, правила изменения объёма и ответственный за управление.

Управление проектами не гарантирует идеальный результат. Но оно не даёт проекту тихо превратиться в бесконечные переделки, споры с заказчиком и сжигание бюджета. Когда в проекте есть понятные роли, планирование, контроль изменений и нормальная коммуникация, довести работу до результата намного проще.

Теперь перейдём к планированию — этапу, где хаос впервые начинают превращать в управляемый проект.

ГЛАВА 2. ПЛАНИРОВАНИЕ БЕЗ ИЛЛЮЗИЙ: КАК ПРЕВРАТИТЬ ИДЕЮ В УПРАВЛЯЕМЫЙ ПРОЕКТ

Что вы узнаете: как превратить устав проекта и верхне-уровневые ожидания в работающую систему управления — от оценки задач, построения графика и расчёта бюджета до планов по рискам, качеству, коммуникациям, приёвке и закупкам.

Планирование — это этап, на котором руководитель проекта впервые вынужден быть неудобным: уточнять, спорить, считать, проверять ожидания и отказываться от красивых иллюзий.

Важно сразу зафиксировать управленческую рамку: планирование — это не разовая активность и не попытка «**угадать будущее**». Это процесс построения модели проекта, которая будет уточняться и пересматриваться по мере появления новой информации.

План — это гипотеза. Управление — это её постоянная проверка и корректировка.

Как вы могли понять из предыдущей главы, этап планирования является одним из наиболее важных в процессе управ-

ления проектом. Именно здесь создаётся дорожная карта проекта, определяются задачи, сроки, бюджет, ресурсы, риски, требования к качеству, порядок коммуникаций, правила приёмки и потребность в закупках.

Многие начинающие руководители проектов на этом этапе ограничиваются только планом-графиком работ. Это опасная ошибка: график без бюджета, ресурсов, рисков, качества, коммуникаций и приёмки создаёт иллюзию управляемости, но не даёт настоящего контроля над проектом.

После изучения этой главы вы сможете увереннее провести этап планирования: разложить проект на задачи, оценить сроки и ресурсы, построить реалистичный график, связать его с бюджетом, заранее учесть риски и подготовить проект к управляемому исполнению.

2.1. НАЗВАНИЕ ПРОЕКТА

Первое, что вы должны сделать на этом этапе — выбрать название проекта. Это может показаться неуместным, но название проекта может иметь большое значение в руководстве. Конечно, легко просто ссылаться на проект, как на «маркетинговый проект» или «проект продаж», но эти названия не дают понимания, что он собой представляет на самом деле. Приведу пару примеров нормальных названий — «Разработка маркетинговой стратегии для увеличения продаж seo услуг» или «Создание стратегии увеличения интернет-трафика». Они гораздо более читабельны, более информативны и помогут лучше понять направление проекта.

2.2. ОПРЕДЕЛЕНИЕ ОБЪЁМА ПРОЕКТА

Определение объёма — основная составляющая и основа для дальнейшей разработки плана проекта. Под объёмом проекта принято понимать определение конечного результата проекта. Главной целью является описание задач с точки зрения заказчика или конечного потребителя.

Неспособность определить содержание проекта указывает на плохое управление изменениями, что может легко привести к провалу проекта.

Работы по определению объёма должен проводить руководитель проекта совместно с проектной командой и привлечением заказчика или его представителя, далее этой группой должны быть согласованы цели и задачи для каждого этапа реализации проекта.

Уточнение:

Определение объёма является достаточно важным этапом работ, все понимают это, как и то, что нечётко сформулированный объём работ становится препятствием в реализации проекта, но понимание этого не мешает даже крупным компаниям пренебрегать этапом определения объёма.

Пора переходить к описанию, что такое «Объём проекта».

2.3. ОБЪЁМ ПРОЕКТА

Объём проекта — письменный документ, используемый всеми членами проектной команды для разработки плана, дальнейшей проработки материалов и оценки результатов проекта. В этом документе подробно и чётко должны быть описаны как минимум несколько факторов, касающихся проекта:

- Какой продукт должен быть передан заказчику по окончании проекта.
- Обоснование проекта.
- Стандарты (могут быть стандарты оформления документации и/или стандарты реализации проекта).
- Цели, исключения, ограничения и прогнозы.
- Описание частей проекта (с указанием ответственных и сроков реализации).

Данный документ служит обязательным приложением к договору или частью договора между заказчиком, исполнителем и всеми остальными заинтересованными лицами. Он обязателен к подписанию и является их обязательством участвовать в проекте.

Замечу:

Информация в документе актуальна на момент его создания. По мере развития проекта данные

могут уточняться, поэтому документ не стоит считать раз и навсегда финальной версией. Корректировки можно вносить на любой стадии проекта через запрос на изменение документации с согласованием заинтересованных сторон.

Из написанного выше можно сделать вывод, что объём проекта — это основа, на которой будут построены все части плана реализации проекта. Чтобы получить полный и правильный документ, нужно собрать ключевую информацию.

Далее рассмотрим подробнее, что нужно собрать и какой должна быть последовательность сбора этой информации:

— **Основная цель.** Делается краткое описание цели проекта, описывается, почему проект является необходимым для заказчика и какие проблемы будет решать продукт (полностью или частично), который получится на выходе проекта. В уставе проекта должно быть предоставлено более подробное описание.

— **Список задач.** Перечень задач верхнего уровня, выполнение задач из этого списка будет происходить на протяжении всего проекта. В этот список также должны войти следующие задачи: составление перечня спецификаций, реализация проекта, тестирование продукта, полученного на выходе проекта.

— **Вехи проекта.** Контрольные или ключевые точки, важные для проекта события, которые будут происходить в определённые моменты времени. Представляются в виде

план-графиков. План-график контрольных точек строится на основе списка задач, получаемых во втором пункте. План-график должен дать представление о первичной (приблизительной) оценке затрат всех существующих видов ресурсов, таких как финансовые и человеческие. Например, тестирование и приём функционала формы обратной связи должны быть окончены до 15 августа текущего года.

— **Технические требования.** Набор определённых требований, которым должен отвечать конечный продукт, чтобы быть качественным. Например, продукт должен быть кроссплатформенным и адаптивным (планшет, смартфон и тому подобное).

— **Исключения.** Определить, что из предполагаемых требований делать не будут, чтобы они не появились потом как новые требования. В любом проекте всегда будет много задач, которые заказчик или его представители захотят добавить в реализацию, но среди них могут быть те, что особо не влияют на продукт и те, что выходят за рамки проекта.

— **Ограничения.** Это важнейшие условия управления проектом, которые находятся в постоянном балансе между собой. К основным ограничениям относятся стоимость, объём и время. В результате их баланса формируется качество.

Такой подход называют «**тройным ограничением**», «**треугольником ограничений проекта**» или «**треугольником управления проектом**» (см. рисунок 2). В вершинах треугольника находятся ключевые ограничения: стои-

мость, объём и время. В центре находится качество как результат их баланса.

Изменение одного ограничения почти всегда влияет на остальные. Если сократить сроки, может вырасти стоимость или уменьшиться объём. Если уменьшить бюджет, могут увеличиться сроки, сократиться объём или пострадать качество. Поэтому проект должен выполняться с учётом этих ограничений.

Далее хочу дать пояснения к условиям (ограничениям) и результату их баланса:

1. **Стоимость** — это бюджет, выделенный для реализации проекта. Он включает общую стоимость ресурсов и возможных затрат.

2. **Объём** — это границы проекта, то есть содержание работ и результатов: что должно быть получено в ходе проекта и что не входит в его рамки.

3. **Время** — это количество доступного времени для завершения проекта, то есть сроки реализации.

4. **Качество** — это совокупность характеристик результата, которые позволяют ему удовлетворять установленные или предполагаемые потребности. В контексте управления проектом таким объектом могут быть результат проекта, отдельные поставляемые материалы, ресурсы проекта или проект в целом.

Указанные ограничения часто соперничают между собой.

Изменение одного из них приводит к изменению других. Например, сокращение сроков может потребовать уменьшения объёма или увеличения стоимости. А уменьшение бюджета может привести к увеличению сроков, сокращению объёма или снижению качества. Эта связь ограничений показана на рисунке 2.



Рисунок 2 — Треугольник управления проектом

Замечу:

— Задача руководителя проекта — не просто знать о существовании этого треугольника, а постоянно управлять балансом между стоимостью, объёмом и временем.

— Если изменения в одном из ограничений не отслеживать и не согласовывать, качество почти неизбежно окажется под ударом. Поэтому руководителю проекта нужна не жёсткость ради жёсткости, а управляемая гибкость.

— **Допущения.** Факторы, оказывающие существенное влияние на проект и являющиеся существенными для достижения целей. Проектная команда не может повлиять на них. Принимается, что они реальны, верны и приняты без доказательств. Например, в случае возникновения ошибки в работе программного комплекса, технический специалист прибудет на объект через 2 часа. Часто путают с рисками, даже опытные и крупные компании.

— **Риски.** Это вероятностные события, которые могут повлиять на проект. Нужно перечислить всё, что может помешать достижению целей, описать возможные последствия, варианты реакции и примерные затраты, если риск наступит. Также стоит указать, как выбранное решение поможет снизить ущерб или обойти проблему.

— **Характеристики приёмки.** Набор критериев, которым должен соответствовать результат проекта. На испыта-

ниях или при проверке должно быть понятно, достигнуты цели проекта или нет.

— **Результаты проекта**. Опишите конкретные результаты проекта.

— **Анализ и приём результатов проекта**. Финальный пункт, на нём происходит совместная проверка результатов работы вместе с заказчиком и прочими заинтересованными лицами. В этом пункте прописывается способ проведения проверки, место, время, кто за что отвечает и участники процесса.

Замечу:

Приведённый выше список можно увеличить или сократить в зависимости от компании и размера проекта.

Уточнение:

Если проект долгосрочный, то нужно обязательно делать долгосрочное планирование. Оно включает всё описанное выше, а также стратегию управления командой и общий подход к развитию проекта. Долгосрочный план обычно охватывает период от трёх лет. Такой план носит описательный характер и определяет общую стратегию компании-разработчика, потому что на таком длинном горизонте трудно предугадать всё заранее.

При составлении такого плана нужно учитывать возможные изменения, новые тенденции и технологии. Уже на его основе

можно делать краткосрочное и среднесрочное планирование, чтобы постепенно достигать промежуточных результатов.

Итак, мы подошли к моменту, когда накопилось много данных. Их нужно систематизировать и превратить в документ, который будет содержать ключевую информацию по проекту, будет согласован и подписан всеми сторонами. Этим документом будет «Устав проекта».

2.4. УСТАВ ПРОЕКТА

Давайте разберёмся, что это за документ такой и для чего он нужен. Приведу одно из существующих определений: **«Устав проекта** — документ, выпускаемый лицом, ответственным за инициацию и запуск проекта в компании-разработчике, формально заявляющий о существовании проекта с предоставлением руководителю проекта полномочия использовать ресурсы организации или привлекать новые для реализации проекта». В Уставе фиксируются бизнес-потребности, допущения, ограничения, понимание потребностей заказчика, высокоуровневые требования, а также описание и название нового продукта, услуги или результат, который планируется создать/получить.

Чаще всего данный документ создаёт руководитель проекта, подписывает его топ-менеджер компании, ответственный за инициацию и запуск проекта.

В этот документ стоит внести ещё ряд пунктов: список документов, материалы и работы, необходимые для реализации продукта, вовлечённость заинтересованных сторон, исключения некоторых возможностей, чтобы в дальнейшем точно определять, лежат ли новые «хотелки» заказчика в пределах границ проекта.

Бывает, что некоторые компании включают в Устав план-график реализации проекта, в этом нет ничего плохого. Но

нужно понимать, что в зависимости от выбранной методологии реализации проекта заказчика, и других факторов, план становится более динамичным, чем сам Устав проекта, который подписывают и редко вносят в него изменения. Поэтому лучшим решением будет вынести план-график реализации проекта из Устава, сделав его отдельным документом, в самом Уставе сослаться на документ «План-график реализации проекта». В Уставе проекта стоит создать раздел «План управления проектом».

План управления проектом — это план, показывающий, как будет происходить мониторинг, контроль и исполнение проекта. «План управления проектом» отличается от «Плана-графика реализации проекта», по сути, это два разных документа и различаются они не только по названию, но и по содержанию.

Замечу:

Устав может быть отдельным документом, а может быть приложением к договору.

Если проект начинается не с готового устава, а только с идеи, полезно сначала коротко упаковать её в продуктовый питч: какую проблему решаем, для кого, какую ценность создаём, какие функции нужны в первую очередь, какие ограничения и риски уже видны. Для этого можно использовать приложение А.7. «Чек-лист продуктового питча».

Далее перейду к описанию структуры и содержания Устава с разбивкой по разделам:

— **Титульный лист.** Вариантов оформления данного раздела много, укажу только обязательную информацию. В разделе должно быть:

— Название организации.

— Название документа (в нашем случае это Устав или Устав проекта).

— Название проекта (например, «Разработка информационно-развлекательного портала»).

— Номер контракта на разработку данного проекта.

— Год создания документа.

— **Оглавление.** Идёт второй страницей. Рекомендую сразу прописать структуру и по ней заполнять документ или при правильном оформлении заголовков автоматически сгенерировать оглавление после заполнения документа.

— **Изменения документа.** В этом разделе нужно фиксировать все изменения документа. В разделе должно быть:

— Версии документа.

— Даты изменения.

— Комментарии.

— Что изменено.

— **Лист согласования.** По сути, это продолжение прошлого раздела и их можно объединить в одну таблицу. Для удобства восприятия рассмотрим его как отдельный раздел. В разделе должно быть:

— Инициатор (кто инициировал изменение).

— Версия согласуемого документа.

- Дата согласования.
- **Сведения о проекте**. В разделе должно быть:
- Полное наименование проекта.
- Краткое наименование проекта.
- Область применения.
- Описание проекта.
- Функциональный заказчик и/или его представители.
- Руководитель проекта.
- Исполнители проекта с указанием ролей и контактных данных.

- Соисполнители (если будут).
- Формальные основания для инициации.
- Взаимосвязь с другими проектами.
- Сроки реализации проекта.
- Автор Устава проекта.
- Дополнительная информация.

— **Цели и задачи проекта**. В этом разделе указывают, какие цели будут достигнуты и какие задачи решены в ходе реализации проекта. В разделе должно быть:

- Наименование цели.
- Наименование задачи (в одной цели их может быть много).
- Метод достижения цели.
- Критерии достижения.
- Статус.
- **Целевые показатели** (результаты). В этом разделе

фиксируют, какие цели и результаты уже достигнуты, какие задачи решены и сколько ещё времени нужно для завершения работ. В разделе должно быть:

- Наименование цели/показателя.

- Текущее значение (или можно сделать стартовое значение).

- Достижение цели: даты проверок (составьте список контрольных точек для проверки показателей).

- **Заинтересованные стороны и их ожидания.** В этом разделе стоит описать функции спонсора проекта, владельца проекта, любых известных поставщиков или внешних экспертов, которые будут использоваться во время реализации проекта. В разделе должны быть указаны:

- Орган/организация.

- Представляющий интересы (ФИО, должность).

- Ожидания от продукта.

- **Реестр рисков.** В разделе нужно описать риски проекта (включая возможные неопределённые события, которые могут возникнуть в проекте и вызвать последствия, влекущие за собой нежелательные эффекты) в виде таблицы. Отдельным блоком нужно описать правила и периодичность пересмотра реестра рисков проекта, всё же это динамичная сущность, как и сам проект. В разделе должны быть указаны:

- Наименование (название риска).

- Ожидаемые последствия.

- Мероприятия по реагированию.

— Вероятность наступления.

— Уровень влияния на проект (например, по шкале от 1 до 5).

— **Допущения и ограничения.** По сути, это внешние и внутренние факторы, предпосылки, на основе которых можно сделать оценку сроков выполнения, трудоёмкости работ проекта и стоимости. В разделе должны быть указаны:

— факторы,

— описание каждого (краткое пояснение).

— **Исключения.** Нужно привести перечень того, что не будет реализовано в функционале продукта и/или то, что не будет сделано в рамках проекта.

— **Контрольные точки.** Список контрольных точек, определяющих ключевые события проекта, их даты и результаты, которые должны быть получены по состоянию на эти даты. Контрольные точки на первых этапах покрывают проектную деятельность (например, собрать требования, разработать документацию, закупить что-то) без реализации. Когда появляется календарный план реализации проекта, то появляются дополнительные контрольные точки, совмещённые со сдачей сборок или окончанием спринта. В разделе должны быть указаны:

— Этапы/события.

— Результаты на этапах.

— Даты наступления (готовности).

— **План проекта.** По сути, нужно расписать фазы жиз-

ненного цикла проекта. Возможна детализация (степень выбирайте для себя самостоятельно). В разделе должны быть указаны:

- Фазы.
- Временные рамки: общие по проекту и отдельно по каждой фазе.
- Содержание (краткий перечень работ в фазе).
- Участники каждой фазы.
- Ответственные за проект и каждую фазу.
- Результат (основные и промежуточные результаты или продукты на выходе фаз).

— **Ключевые роли и функциональные обязанности участников проекта.** В разделе кратко раскрываются ключевые роли и функциональные обязанности всех участников проекта. Оформить можно в виде списка или таблицы. В разделе должны быть указаны:

- роли (с указанием списка людей),
- обязанности (по ролям).
- **Документы по проекту, требующие согласования и утверждения.** В данном разделе нужно собрать все документы, которые нужно будет разработать, согласовать и утвердить в рамках проекта. В разделе должны быть указаны:

- Название документа.
- Разработчик (указать, кто готовит документ, роль).
- Утверждает (указать, кто утверждает документ, роль).

Последней страницей документа будет пространство для подписей всех, кто связан с проектом.

Много всего в Уставе, но это только кажется, не переживайте. Кроме Устава должен быть ещё ряд документов для управления проектом:

- Поэтапные планы работ (про общий план-график реализации проекта писал ранее).
- Бюджетный план проекта.
- Протоколы статусных совещаний.
- Протоколы собраний рабочих групп.
- Отчёты о состоянии проекта.

После создания документа «Устав» и подписания его, нужно переходить к планированию ресурсов и их оценке.

2.4.1. ОТ УСТАВА К РЕАЛЬНОСТИ: КАК РАЗЛОЖИТЬ ПРОЕКТ НА ЗАДАЧИ И НЕ СОВРАТЬ СРОКАМИ

Устав проекта фиксирует верхнеуровневую договорённость: зачем нужен проект, какой результат ожидается, кто является ключевыми участниками, какие есть ограничения и в каких примерных рамках проект должен быть выполнен.

Но устав — это ещё не план работ.

Устав отвечает на вопрос: «**Что мы в целом хотим получить и зачем?**»

Планирование должно ответить на другие вопросы:

— «Что конкретно нужно сделать?»

— «Кто будет это делать?»

— «Сколько времени это займёт?»

— «Какие ресурсы понадобятся?»

— «Что может пойти не так?»

— «Как мы поймём, что результат готов?»

Именно здесь начинается переход от идеи к управляемому проекту.

После подписания устава часто появляется опасная иллюзия: «**Ну всё, вроде договорились, теперь можно делать**». На практике это не так. Устав задаёт направление,

но не заменяет декомпозицию, оценки, зависимости, допущения, план-график и критерии приёмки.

Фразы вроде:

- «примерно понятно»;
- «потом уточним»;
- «там ничего сложного»;
- «команда разберётся по ходу»;
- «это же очевидно»;
- «заказчик потом объяснит подробнее»

почти всегда означают будущие проблемы со сроками, бюджетом, качеством или ожиданиями заказчика.

Задача руководителя проекта — не позволить проекту стартовать на туманных формулировках. Чем раньше неопределённость будет превращена в задачи, допущения, ограничения, оценки и критерии готовности, тем выше шанс, что проект останется управляемым.

РАЗЛОЖИТЕ РЕЗУЛЬТАТ НА ПОНЯТНЫЕ ЧАСТИ

Первое, что нужно сделать после устава, — определить, из каких крупных частей состоит будущий результат.

Если проект связан с разработкой продукта, это могут быть функциональные блоки, пользовательские сценарии, интеграции, отчёты, роли пользователей, административные функции, настройки, документация и обучение.

Если проект связан с внедрением системы, это могут быть обследование, настройка, миграция данных, интегра-

ции, обучение пользователей, тестовая эксплуатация и запуск.

Если проект организационный, это могут быть регламенты, процессы, роли, обучение сотрудников, коммуникации, контрольные точки и итоговые документы.

На этом шаге не нужно сразу оценивать всё в часах. Сначала нужно понять, из чего вообще состоит проект.

Хороший вопрос для команды: **«Какие крупные результаты должны появиться, чтобы заказчик мог сказать: проект выполнен?»**.

После этого крупные результаты нужно постепенно разложить на более мелкие управляемые задачи.

ОТЛИЧАЙТЕ ЗАДАЧУ ОТ РАЗМЫТОГО НАМЕРЕНИЯ

Одна из типичных ошибок начинающего руководителя проекта — включать в план слишком крупные или размытые формулировки.

Плохо:

- «сделать интеграцию»;
- «разработать личный кабинет»;
- «настроить отчёты»;
- «подготовить документацию»;
- «обеспечить качество»;
- «согласовать с заказчиком».

Такие формулировки выглядят как задачи, но на самом

деле они слишком общие. Их трудно оценить, трудно контролировать и трудно принять.

Лучше:

- описать требования к интеграции;
- согласовать протокол обмена данными;
- подготовить тестовые данные;
- реализовать отправку запроса;
- реализовать обработку ответа;
- обработать ошибки интеграции;
- провести интеграционное тестирование;
- подготовить инструкцию по эксплуатации;
- провести демонстрацию заказчику.

Управленческое правило:

Если задачу нельзя поставить конкретному исполнителю, оценить по времени и принять по понятному результату — она ещё не готова для плана-графика.

Это правило помогает быстро отделить настоящие задачи от красивых, но бесполезных формулировок.

НЕ ДЕЛАЙТЕ ЗАДАЧИ СЛИШКОМ БОЛЬШИМИ

Если задача слишком большая, она превращается в «чёрный ящик». Вроде бы работа идёт, исполнитель занят, статус стоит «в работе», но руководитель проекта не понимает, что именно происходит внутри.

Опасные признаки слишком крупной задачи:

- задача оценивается в несколько недель;
- внутри неё участвуют разные специалисты;
- результат нельзя проверить одним действием;
- исполнитель не может кратко объяснить, что будет готово на выходе;
- у задачи нет понятного критерия завершения;
- задача регулярно переносится без ясной причины.

Практическое правило:

Если задача занимает больше 8—16 часов чистой работы, стоит проверить, нельзя ли разбить её на меньшие части.

Это не жёсткий закон. В некоторых проектах задачи могут быть крупнее. Но для начинающего РП такое правило помогает не потерять управляемость.

Мелкая задача лучше видна, легче оценивается, быстрее проверяется и проще передаётся между участниками.

ОПРЕДЕЛИТЕ РЕЗУЛЬТАТ КАЖДОЙ ЗАДАЧИ

У каждой задачи должен быть понятный результат. То есть:

- Не просто «аналитик занимался требованиями», а «подготовлен и согласован раздел требований по регистрации пользователя».
- Не просто «разработчик делал интеграцию», а «реализована отправка заявки во внешнюю систему и обработка успешного ответа».
- Не просто «тестировщик проверял модуль», а

«проведено тестирование сценариев регистрации, авторизации и восстановления пароля, заведены дефекты».

Если результат задачи нельзя описать одним-двумя предложениями, задача, скорее всего, сформулирована плохо.

Для каждой важной задачи полезно ответить на вопросы:

- что должно быть готово на выходе;
- кто принимает результат;
- по каким критериям результат считается выполненным;
- какие материалы или данные нужны для начала работы;
- какие зависимости есть от других задач или людей.

Так руководитель проекта постепенно превращает абстрактную идею в управляемый набор работ.

ПРОВЕРЬТЕ ЗАВИСИМОСТИ И ДОПУЩЕНИЯ

После декомпозиции нужно понять, какие задачи зависят друг от друга.

Например:

- нельзя тестировать функцию, пока она не разработана;
- нельзя разработать интеграцию, пока не согласован протокол обмена;
- нельзя запустить обучение пользователей, пока не готова инструкция;
- нельзя подписать акт приёмки, пока не закрыты кри-

тические замечания;

— нельзя начать миграцию данных, пока не утверждены правила очистки и переноса.

Зависимости важны потому, что они влияют на порядок работ и сроки проекта. Если не увидеть зависимости заранее, проект начнёт тормозить уже на исполнении: одни участники будут ждать входные данные, другие — согласования, третьи — доступы, четвёртые — результаты предыдущих задач.

Хорошие вопросы **«Что должно быть готово до того, как мы начнём эту задачу?»** и **«Кто будет заблокирован, если эта задача задержится?»**.

Кроме зависимостей, нужно зафиксировать допущения. Допущения — это условия, при которых план считается реалистичным. Например:

- заказчик будет отвечать на вопросы в течение двух рабочих дней;
- состав команды не изменится;
- ключевые специалисты будут доступны в согласованном объёме;
- требования не будут расширяться без процедуры изменения;
- внешняя система будет доступна для интеграции;
- подрядчик передаст материалы в срок;
- тестовая среда будет готова к нужной дате;

— согласующие лица со стороны заказчика не изменятся.

Допущения — это не слабость планирования. Это честное описание условий, при которых план может быть выполнен.

Проблема начинается тогда, когда допущения существуют только в голове руководителя проекта или команды. Если допущения не зафиксированы, график становится фикцией. Вроде бы срок согласован, но никто не проговорил, что он возможен только при своевременной обратной связи заказчика, стабильном составе команды и отсутствии новых требований. Пример формулировки:

«План-график рассчитан при условии, что заказчик предоставляет обратную связь по переданным материалам в течение двух рабочих дней. При увеличении срока согласования общий срок проекта может быть пересмотрен».

Такая фраза защищает проект от ситуации, когда задержки возникают из-за внешних ожиданий, но ответственность за них автоматически пытаются переложить на команду.

ОТДЕЛИТЕ НЕОБХОДИМОЕ ОТ ЖЕЛАТЕЛЬНОГО

На этапе планирования важно понять, что действительно нужно для первого полезного результата, а что можно отложить.

Хороший вопрос **«Без чего результат вообще не имеет смысла, а что можно добавить позже?»**.

Например, для первой версии личного кабинета критиче-

ски важны регистрация, вход, профиль и базовая работа с заявками. А сложная аналитика, красивые персональные настройки и расширенные уведомления могут быть вынесены во второй этап.

Если этого не сделать, проект легко превращается в бесконечную попытку реализовать всё сразу.

Важно объяснить заказчику: перенос части функций во второй этап — это не отказ от них, а способ быстрее получить работающий результат и снизить риски.

СВЯЖИТЕ ЗАДАЧИ С ОТВЕТСТВЕННОСТЬЮ

Когда проект разложен на задачи, у каждой задачи должен появиться ответственный.

Ответственный — это не обязательно единственный исполнитель. Это человек, который отвечает за то, чтобы задача была доведена до результата: понял требования, уточнил вопросы, выполнил работу или организовал выполнение, сообщил о проблемах и передал результат дальше. Приведу примеры плохих и хороших определений ответственных:

— Так плохо **«Команда разработки»**, а так лучше **«Иван, Backend-разработчик»**.

— Так плохо **«Аналитики»**, а так лучше **«Мария, аналитик, отвечает за согласование требований по интеграции»**.

Чем яснее ответственность, тем меньше риска, что задача «повиснет между людьми».

Руководитель проекта должен помнить: перед заказчиком

за общий результат отвечает не разработчик, не аналитик и не тестировщик. Перед заказчиком отвечает руководитель проекта. Поэтому РП должен видеть не только список задач, но и то, кто отвечает за каждую важную часть результата.

ПРОВЕРЬТЕ РЕАЛИСТИЧНОСТЬ ПЕРЕД ОЦЕНКОЙ СРОКОВ

Перед тем как переходить к оценке сроков, нужно проверить, достаточно ли задача понятна всем. Для этого каждой важной задаче стоит задать несколько вопросов:

- понятно ли, что именно нужно сделать;
- есть ли критерий готовности;
- понятно ли, кто исполнитель;
- известны ли зависимости;
- есть ли входные данные;
- есть ли внешние ограничения;
- понятно ли, кто принимает результат;
- есть ли риски, которые могут повлиять на выполнение.

Если на эти вопросы нет ответов, точная оценка сроков будет иллюзией.

В такой ситуации правильнее не давать уверенную оценку, а зафиксировать неопределённость и уточнить недостающие данные.

Как это должно выглядеть:

Плохо «**Сделаем за неделю**», а так лучше «**Предварительно — около недели, если внешний API работа-**

ет по описанной документации и не потребуется доработка со стороны подрядчика. После проверки API оценку уточним».

Так руководитель проекта не «страхуется от всего», а честно показывает условия, при которых срок реалистичен.

ИТОГ

Переход от устава к плану — это момент, где проект либо становится управляемым, либо начинает копить будущие проблемы.

Если после устава сразу перейти к красивому графику, не разобрав объём, задачи, зависимости, допущения, ответственность и критерии готовности, такой график будет выглядеть убедительно, но быстро начнёт разрушаться на этапе исполнения.

Хороший план начинается не с дат. Хороший план начинается с ясности.

Запомните:

- устав проекта — это не план работ;*
- крупные результаты нужно разложить на управляемые задачи;*
- задача должна иметь исполнителя, результат и критерий готовности;*
- слишком крупные задачи скрывают риски;*
- зависимости нужно выявлять до построения графика;*
- допущения нужно фиксировать письменно;*

— *желательное нужно отделять от необходимого;*

— *перед заказчиком за общий результат отвечает руководитель проекта.*

Дальше начинается самая практическая часть планирования. Мы переходим к вопросу, который быстро возвращает любого руководителя проекта к реальности: кто будет делать работу, сколько времени это займёт, какие ресурсы нужны и где план может сломаться.

Именно здесь проект перестаёт быть красивой идеей и начинает превращаться в управляемую систему сроков, людей, денег, рисков и договорённостей.

2.5. ПЛАНИРОВАНИЕ И ОЦЕНКА РЕСУРСОВ

Для начала стоит ответить себе на следующие вопросы:

- На что компания тратит больше всего денег?
- Что является для компании главным ресурсом?

Ответ на оба вопроса один, он прост и думаю вы сами это понимаете — **сотрудники компании**. Сотрудники, которые своим трудом создают продукты и зарабатывают для компании деньги. И именно на их зарплаты, отпуска, больничные, налоги, премии, обучение, оборудование и прочее приходится основная масса затрат компании.

Планирование этих ресурсов позволяет получить нужное количество ресурсов в нужный момент и избежать незанятости («простоя») сотрудников, а это значит минимизировать потери.

Итак, давайте теперь поговорим о самом ресурсном планировании. Начнем с того, что такое планирование ресурсов.

2.5.1. ПЛАНИРОВАНИЕ РЕСУРСОВ

Планирование ресурсов — это процесс определения того, какие ресурсы необходимы для успешного выполнения различных видов деятельности в рамках проекта, как, когда, на что эти ресурсы будут выделены и использованы.

Хороший план ресурсов включает подробную информацию о каждом типе ресурсов, необходимом для проекта, включая детализацию по отдельным членам проектной команды для решения конкретных задач.

Наиболее распространёнными категориями ресурсов являются:

- люди;
- знания;
- деньги;
- техническое оборудование;
- время;
- материалы.

Планирование ресурсов — это не список людей, оборудования и денег. Такой список можно сделать за десять минут. Проблема в другом: сможет ли проект реально получить эти ресурсы тогда, когда они понадобятся.

Руководителю проекта нужно понять не только **какие ресурсы нужны**, но и **можно ли будет реально использо-**

вать их в нужный момент.

Например, человек может быть указан в проекте, но фактически быть занят другим направлением. Оборудование может существовать в компании, но использоваться другим проектом. Нужный эксперт может быть доступен только на короткую консультацию. Доступ к системе может быть нужен уже сегодня, но выдан только через неделю.

Поэтому планирование ресурсов должно отвечать не только на вопрос «**Что нам нужно?**», но и на вопросы:

- «Когда это понадобится?».
- «Кто подтверждает доступность ресурса?».
- «Есть ли ограничения по использованию ресурса?».
- «Что произойдёт, если ресурс не будет доступен вовремя?».

Если эти вопросы не задать заранее, ресурсный план легко превращается в список пожеланий.

А список пожеланий — это не план. Это аккуратно оформленная надежда.

РЕСУРС — ЭТО НЕ ТОЛЬКО ЧЕЛОВЕК

Когда говорят о ресурсах, чаще всего сразу думают о людях. Это логично: именно специалисты выполняют основную работу и создают результат проекта.

Но ресурсами проекта являются не только сотрудники.

Ресурсом может быть:

- специалист с нужной компетенцией;

- эксперт, который нужен на несколько часов для консультации;
- подрядчик;
- оборудование;
- сервер;
- тестовая среда;
- лицензия на программное обеспечение;
- помещение;
- материалы;
- доступ к данным;
- согласование от конкретного должностного лица;
- время заказчика или представителя бизнеса.

Иногда проект задерживается не из-за отсутствия команды, а из-за отсутствия одного небольшого, но критичного ресурса: доступа, лицензии, тестового стенда, подписи, эксперта или оборудования.

Поэтому при планировании ресурсов важно задавать не только вопрос **«Кто будет делать задачу?»**, нужно также спрашивать: **«Что ещё нужно, чтобы эта задача действительно могла быть выполнена?»**.

Например, для задачи «провести тестирование» может понадобиться не только тестировщик, но и готовая сборка продукта, тестовая среда, тестовые данные и доступы.

Если чего-то из этого нет, задача в плане есть, но реальной возможности выполнить её нет.

Это одна из частых ловушек ресурсного планирования: задача записана, исполнитель назначен, а работать всё равно невозможно.

НАЛИЧИЕ РЕСУРСА И ДОСТУПНОСТЬ РЕСУРСА

Важно различать наличие ресурса и его доступность:

— Наличие означает, что ресурс в принципе существует.

— Доступность означает, что ресурс можно использовать в нужный период и в нужном объёме.

Разница очевидна.

Специалист может быть в штате, но не быть доступным проекту в нужные даты. Оборудование может быть в компании, но уже использоваться другой командой. Бюджет может быть согласован, но деньги фактически станут доступны только после отдельного утверждения. Заказчик может быть назначен ответственным за проверку, но реально отвечать на вопросы раз в неделю.

Плохой ресурсный план говорит — **«Нам нужен аналитик»**.

Хороший ресурсный план уточняет — **«Нам нужен аналитик в период подготовки требований, и его участие должно быть подтверждено заранее»**.

Плохой ресурсный план говорит — **«Нужна тестовая среда»**.

Хороший ресурсный план уточняет — **«Тестовая среда должна быть готова до начала проверки, с выданными»**

ми доступами и тестовыми данными».

Вот здесь и проходит граница между планом и красивой самообманной таблицей.

Ресурс в таблице не равен ресурсу в проекте. Пока доступность не подтверждена, это не ресурс, а предположение.

КОМПЕТЕНЦИИ РЕСУРСОВ

Ещё одна ошибка — считать ресурсы полностью взаимозаменяемыми.

Два специалиста с одной должностью не всегда могут одинаково быстро и качественно выполнить одну и ту же задачу. Поясню на примере:

- один разработчик хорошо знает нужный модуль, другой видит его впервые;
- один аналитик уже работал с этим заказчиком, другой только входит в контекст;
- один тестировщик понимает бизнес-логику продукта, другой может проверить только формальные сценарии;
- один подрядчик умеет работать в нужных сроках, другой требует постоянного контроля.

Поэтому при планировании ресурсов нужно учитывать не только роль, но и реальные компетенции. Недостаточно будет написать: **«Назначить разработчика»**, лучше сразу уточнить: **«Нужен Backend-разработчик, который знает интеграцию с платёжным сервисом»**. Или недостаточно написать: **«Нужен аналитик»**, лучше: **«Нужен аналитик, который понимает бизнес-процесс заказчика и**

сможет быстро согласовать требования с пользователями».

Вот мы и приходим к пониманию, что проекту нужны не абстрактные **«единицы ресурсов»**, а конкретные знания, опыт, доступы и возможности.

Если не учитывать компетенции, можно формально закрыть потребность в ресурсах, но фактически получить задержки, ошибки и переделки.

В документе всё будет выглядеть прилично. В работе — начнёт сыпаться.

ЧТО ДОЛЖЕН ПОКАЗАТЬ ПЛАН РЕСУРСОВ

Хороший план ресурсов должен помогать руководителю проекта ответить на несколько практических вопросов:

- какие ресурсы нужны проекту;
- для каких задач они нужны;
- когда они понадобятся;
- кто отвечает за их предоставление;
- подтверждена ли их доступность;
- какие ресурсы являются критичными для старта или продолжения работ.

Последний пункт особенно важен.

Если без конкретного специалиста, среды, оборудования, доступа или согласования проект остановится, это уже не просто ресурс. Это ограничение проекта, которое нужно контролировать заранее.

Плохое ресурсное планирование часто можно узнать по простой фразе: **«Мы думали, он будет доступен»**. Но реальность другая, хорошее ресурсное планирование заранее уточняет: **«Он действительно доступен? Кто это подтвердил? В какие даты? Какие есть ограничения? Что делаем, если он недоступен?»**.

Пока на эти вопросы нет ответов, неопределённость высока и ресурсный план нельзя считать готовым.

ПРАКТИЧЕСКАЯ ПРОВЕРКА РЕСУРСНОГО ПЛАНА

Перед переходом к оценке ресурсов план нужно проверить на реальность. Не **«пробежать глазами»**, а именно **проверить**.

Для этого можно пройтись по простому списку вопросов:

- все ли ключевые ресурсы перечислены;
- понятно ли, зачем нужен каждый ресурс;
- указано ли, когда ресурс понадобится;
- подтверждена ли доступность ключевых специалистов;
- есть ли у специалистов нужные компетенции;
- нужны ли доступы, лицензии, оборудование или тестовые среды;
- понятно ли, кто отвечает за предоставление ресурса;
- есть ли зависимость от заказчика, подрядчика или другого подразделения;
- понятно ли, что произойдёт, если ресурс не будет доступен вовремя.

Если хотя бы по одному ключевому ресурсу ответ неясен, значит ресурс ещё не спланирован до конца.

Он может быть указан в документе, но пока не подтверждён управленчески.

Неподтверждённый ресурс лучше сразу пометить как **ограничение** или **потенциальную проблему**. Иначе он всплывёт позже — обычно в самый неудобный момент. В проектах такие вещи редко всплывают красиво и заранее. **Обычно они всплывают тогда, когда уже горят сроки.**

ПЕРЕХОД К ОЦЕНКЕ РЕСУРСОВ

После того как руководитель проекта определил виды деятельности и необходимые ресурсы, можно переходить к следующему шагу — оценке ресурсов.

Важно не путать эти два шага:

— Планирование ресурсов отвечает на вопрос: «**Какие ресурсы нужны проекту и когда они должны быть доступны?**».

— Оценка ресурсов отвечает на вопрос: «**Сколько этих ресурсов потребуется и сколько времени займёт работа с их учётом?**».

Например, на этапе планирования ресурсов руководитель проекта определяет, что для задачи нужны аналитик, тестировщик, тестовая среда, доступы и участие представителя заказчика.

А на этапе оценки ресурсов он уже уточняет объём, дли-

тельность использования и ограничения по каждому ресурсу. Это следующий шаг, который разберём отдельно.

Если перепутать эти шаги, можно начать считать сроки до того, как стало понятно, есть ли вообще нужные люди, знания, оборудование, доступы и условия для выполнения работы.

Получится красивая оценка на слабом фундаменте. А слабый фундамент обычно не спрашивает разрешения перед тем, как треснуть.

ИТОГ

Планирование ресурсов — это не просто список людей и материалов. Это проверка того, сможет ли проект получить нужные ресурсы в нужный момент и использовать их по назначению.

Руководитель проекта должен учитывать не только наличие ресурсов, но и их доступность, компетенции, ограничения и зависимости.

Запомните:

— ресурс — это не только человек, но и знания, деньги, оборудование, время, материалы, доступы и согласования;

— наличие ресурса ещё не означает его доступность;

— ресурс должен быть нужен не «вообще», а под конкретную задачу и период;

— компетенции важны не меньше количества

людей;

— ресурс в таблице без подтверждённой доступности — это не ресурс, а надежда;

— хороший ресурсный план показывает не только что нужно, но и кто, когда и на каких условиях это предоставит;

— сначала нужно понять, какие ресурсы нужны и доступны, а затем переходить к их оценке.

2.5.2. ОЦЕНКА РЕСУРСОВ

По моему мнению может быть один вариант оценки ресурсов — через составление объёма работ и его оценку.

Оценки задач могут измеряться в часах, днях, неделях и месяцах, в зависимости от нужной глубины детализации. Чаще всего для измерения выбирают часы.

Задачи будут вписаны в план-график, чтобы свести к минимуму пересечения («перекрытия») задач и ограничения. Оценка продолжительности должна учитывать саму работу, её масштабы и наличие необходимых ресурсов.

Также важно сбалансировать время и ресурсы членов команды, которые планируется назначить для работы над задачами, понимая, что чем больше ресурсов, тем меньше времени может потребоваться для реализации.

Замечу:

Описанное выше правило работает не всегда.

То есть не всегда пропорциональное увеличение числа исполнителей ведёт к пропорциональному уменьшению времени реализации, влияет на это ряд факторов, приведу некоторые критичные, оказывающие наиболее сильное влияние:

- Профессионализм исполнителей.
- Участие выделенных на проект специалистов в других проектах.
- Невозможность распараллелить задачи.

Ещё один важный момент для руководителя проекта — стоимость каждого ресурса. Исходя из своего опыта скажу, что она складывается минимум из: **оплаты труда** (зарплата, подоходный налог, премии — да, это нужно учитывать, выплаты в пенсионный фонд и фонд социального страхования, медицинская страховка, иногда компенсация питания), **общехозяйственных** и **административных расходов** (рабочее место, оборудованное помещение, техника, связь, серверы, обеспечивающий персонал — HR, бухгалтеры, юристы, персонал обслуживания — уборщики, завхозы). Могут быть и другие пункты. Эта информация понадобится для следующего шага в планировании. Итак, с общей вводной закончили, идем дальше.

В начале проекта по категориям ресурсов известно не всё, и это нормально. Они будут уточняться по мере того, как руководитель проекта собирает данные и прорабатывает, что нужно сделать для достижения конечного результата. По мере детализации и разделения отдельных мероприятий появится возможность организовать их в последовательные группы.

Для первичного ресурсного планирования достаточно получить список того, что хочет клиент. Это также даст возможность спрогнозировать объём работ, количество специалистов, последовательность работ. Если клиент предоставляет этот список, то это очень хорошо. Часто бывает, что

клиент не до конца сам понимает, что он хочет и нужно самим сделать за него эту работу, и согласовать с ним. Приведу несколько методов, которые может использовать руководитель проекта для определения возможного функционала планируемого продукта:

— **Привлечение эксперта.** В этом варианте руководитель проекта консультируется с экспертом, который ранее выполнял работу в области, связанной с интересующим проектом. Эксперт, как правило, может дать большой совет о том, с чего начать с точки зрения возможного функционала и требуемых на реализацию ресурсов.

— **Анализ альтернатив.** В этом варианте руководитель проекта и проектная команда проводят мозговой штурм, результатом которого будет примерный список возможностей проекта и требуемых ресурсов для реализации. Такой подход часто даёт на выходе ряд вариантов функциональных возможностей с различными трудозатратами, естественно, выбираются оптимальные варианты, остаются два или три варианта — самый дешёвый, самый дорогой и/или комбинация.

— **Опубликованные данные.** В этом варианте руководитель проекта обращается к научным статьям и другим публикациям, которые позволяют собрать информацию о подобных проектах, их возможностях и возможно цене.

— **Оценка снизу вверх.** В этом варианте проектная команда разбивает проект на отдельные виды деятельности, где

они оценивают ресурсы для каждой части, а затем складывают всё вместе для полной оценки (с учётом возможных рисков, конечно). Хотя этот метод является наиболее точным из всех, он может быть чрезвычайно трудоёмким, а значит, и дорогостоящим.

— **Комбинация.** Совместное использование нескольких вариантов из вышеуказанных.

В своей работе часто использую комбинированный подход, состоящий из «**Анализ альтернатив**» и «**Оценка снизу вверх**», на нём и сосредоточусь.

Замечу:

На более мелкие временные интервалы, чем неделя делать разбивку при первом приближении не стоит. Со временем план будет меняться и детализироваться.

Процесс оценки можно разбить на шаги. У меня их семь, у вас может что-то отличаться. Далее распишу эти шаги и что в них нужно делать:

— **Шаг 1.** Допустим, мы получили от заказчика описание, какой продукт он хочет и какие задачи он должен решать. Клиент заполнил Бриф (сам или совместно с руководителем проекта), на его основе сформировали и проработали бизнес-требования. Далее проектная команда под управлением руководителя проекта прогнозирует (метод мозгового штурма), какой функционал может потребоваться заказчику (нужно считать максимальный вариант возможностей,

главное, чтобы это было логично, а не просто добавили для объёма) для достижения его целей.

— **Шаг 2.** Когда проектная команда сформировала у себя видение проекта, нужно систематизировать информацию, превратив её в список функциональных задач. Для этого также нужны усилия руководителя проекта и проектной команды. Список функциональных задач будет выглядеть приблизительно так:

- Регистрация и управление пользователями.
- Создание и управление группами.
- Создание онлайн-заявки на клиентской части (форма заявки с такими-то параметрами...).
- Управление расписанием сотрудников.
- Планирование закупок.
- Ведение и просмотр статистики по заявкам (перечисляются параметры и фильтры). — И так далее.

Если есть необходимость, то функциональные задачи можно разбить на подзадачи.

Этот список желательно согласовать с клиентом, чтобы понять, правильно ли его поняли и нужно ли ему то, что описано в списке. Некоторые пропускают это согласование и делают сразу прогнозную оценку работ по задачам, собирая данные в отдельный документ (тут нужно понимать проблемный клиент или нет, можно себе такое позволить или будут проблемы). Этот документ создаётся на этапе «Ини-

циация», его также называют ballpark, о нём мы говорили в первой главе.

— **Шаг 3.** Далее подробнее опишу, что это за документ и как это всё должно работать. Руководитель проекта с проектной командой делают прогнозную оценку по времени разработки предполагаемого функционала (с максимально допустимыми возможностями, но в пределах разумного).

Следующий шаг — объединить задачи в блоки. Например, прототипирование, разработку дизайна и вёрстку можно собрать в блок «Дизайн и вёрстка». Для каждого блока нужно указать, какие специалисты участвуют в работе, сколько времени им потребуется и какая ставка используется для расчёта.

Если над одним блоком работают разные типы специалистов, лучше разделить их по строкам: отдельно дизайнер, отдельно верстальщик, отдельно разработчик и так далее. После этого считаем стоимость каждого блока и общую стоимость проекта. Ниже, в таблице 2, приведён пример такой таблицы с прогнозными расчётами.

Описание	Позиция	Объем	Единица измерения	Стоимость единицы (руб. до НДС)	Итого (руб. до НДС)
Разработка дизайна (включая адаптивную версию)					
Прототипирование и дизайн (разработка общего стилового решения: сетка, пропорции, размеры и форма блочных элементов)	Арт-директор	50,00	час	1000,00	50 000,00р.
	UX дизайнер	20,00	час	1000,00	20 000,00р.
	Дизайнер	80,00	час	1000,00	80 000,00р.
	Копирайтер	10,00	час	1000,00	10 000,00р.
Стоимость блока:		160 000,00р.			
Вёрстка обычной версии					
Вёрстка обычной версии сайта (Desktops (≥1200px))	Программист (front-end)	40,00	час	1000,00	40 000,00р.
	UX дизайнер	2,00	час	1000,00	2 000,00р.
Стоимость блока:		42 000,00р.			
Вёрстка адаптивной версии					
Вёрстка адаптивной версии сайта (Extra small(<768px), Small devices (≥768px), Medium (≥992px))	Программист (front-end)	100,00	час	1000,00	100 000,00р.
	UX дизайнер	3,00	час	1000,00	3 000,00р.
Стоимость блока:		103 000,00р.			
Интеграция с CMS					
Интеграция вёрстки в систему управлени CMS, разработка функционала, настройка модулей	Программист (back-end)	40,00	час	1000,00	40 000,00р.
	Программист (front-end)	30,00	час	1000,00	30 000,00р.
Стоимость блока:		70 000,00р.			
Прочие работы					
Проверка, тестирование и администрирование, покупка редакции CMS	Технический директор	2,00	час	1000,00	2 000,00р.
	Тестировщик	63,00	час	1000,00	63 000,00р.
	Программист (back-end)	5,00	час	1000,00	5 000,00р.
	Программист (front-end)	4,00	час	1000,00	4 000,00р.
	Системный админ.	1,00	час	1000,00	1 000,00р.
	Контент редактор	30,00	час	1000,00	30 000,00р.
	Лицензия CMS	1,00	шт	5000,00	5 000,00р.
Стоимость блока:		110 000,00р.			
Менеджмент					
Управление проектом, координация работ, взаимодействие с клиентом, работа с докум-и	Менеджер проектов	40,00	час	1000,00	40 000,00р.
	Аккаунт-менеджер	20,00	час	1000,00	20 000,00р.
Стоимость блока:		60 000,00р.			
Итого без НДС:		545 000,00р.			
НДС 20%:		109 000,00р.			
Итого с НДС:		654 000,00р.			

Таблица 2 — Пример таблицы с прогнозными расчётами

В этом примере нет всех блоков (только связанные с разработкой продукта), нет детальной разбивки на задачи, не показаны явно заложенные риски (заложил их в оценку, без

вынесения в отдельную колонку), но для данного примера они не нужны, это пример оформления приблизительной стоимости разработки продукта. Разбивка и детализация здесь не ограничены, можно разбить блоки на более мелкие части или задачи, тут как удобно, как договоритесь с заказчиком или в своём коллективе.

Этот документ показывает заказчику стоимость разработки продукта: какой функционал вы предлагаете и за какую цену. Чаще всего к подобной смете прикладывают документ с описанием планируемого продукта — дерево проекта (как вид структуры) с указанием и кратким описанием функционала в разделах, но без сильной детализации. Для себя вы можете составить вариант с подробным описанием, чтобы зафиксировать и не забыть предложенные идеи.

Если заказчик не согласовал стоимость, нужно подумать, за счёт чего можно уменьшить объём работ, а значит и стоимость: упростить часть функционала, убрать отдельные возможности или увеличить срок работы над проектом, если такой вариант подходит клиенту.

— **Шаг 4.** После достижения согласия по стоимости работ и списку функциональных возможностей между всеми сторонами стоит приступить к детальному описанию функциональных возможностей (с постоянным взаимодействием с заказчиком) и проработке технического задания.

Замечу:

— *Не стоит пренебрегать этапом*

прототипирования, он важен для формирования структуры проекта и понимания у заказчика чего он хочет.

— Нужно согласовывать промежуточные версии разрабатываемой документации.

— **Шаг 5.** Итак, мы имеем разработанное техническое задание и все дополнительные данные. С этими материалами ничего не мешает сделать разбивку на функциональные блоки, а их в свою очередь разделить на задачи. Старайтесь, чтобы задачи не превышали по продолжительности 8 часов. Да, не всегда сделать это возможно, но подумайте, возможно ли разбить на несколько задач или сделать в ней несколько подзадач, если нет, то оставляйте. Это нужно будет в будущем для удобного отслеживания прогресса разработки задач с меньшим шагом ожидания, а значит будет больше шансов вовремя принять меры при возникновении проблем.

— **Шаг 6.** После следует сделать переоценку, но уже с учётом возможных рисков и корректировок от заказчика, тестирования (лучше не объединять с часами разработчиков, а отдельной строкой указать в задаче) и конкретной команды. Команда прорабатывает задачи, оценивая каждую, чтобы получить более точные сроки работ и трудозатрат.

Например, есть задача, **программист оценил время работы** над ней в 10 часов, **сверху закладываем риски 15%** (в зависимости от сложности проекта эту цифру можно увеличить или уменьшить, но меньше 10% закладывать не сто-

ит), закладываем 30% на **тестирование и корректировки**. Если перевести в формулу, то получаем следующее — $(10ч + 15\%) + 30\% = 15ч$. Солидная разница, не правда ли? А теперь представьте, что будет, если задач много и проект от двух-трёх месяцев, какие тогда будут погрешности (о проектах от года даже не говорю), а это именно то время, которого не хватит при работе над проектом.

Итак, у руководителя проекта появилось чёткое понимание, сможет ли проектная команда реализовать планируемый функционал данными ресурсами или нет. Если нет, то мы смотрим, сколько нужно ещё специалистов и каких, корректируем планируемое число участников команды (если это возможно). Если нет, временно оставляем расчёты как есть и отдельно фиксируем нехватку ресурсов. Следующее, что нужно сделать, это проанализировать, какие задачи требуют много времени и уже после этого принимать решение об оптимизации объёмов, ресурсов и самого расписания.

Так как руководитель проекта уже получил оценку по каждой задаче от каждого специалиста с заложенными рисками, можно сводить эти данные к общему времени, необходимому на решение каждой задачи. Не забудьте удостовериться перед началом этих работ, что у всех членов команды есть актуальная версия технического задания и список задач, созданный ранее проектной командой, кто-то может что-то упустить или забыть. Выдача материалов команде — хороший вариант, так как вы получите дополнительную прора-

ботку материалов проектной командой, уточнение или разбивку на дополнительные задачи.

Наименование	Описание	Исполнитель	Оценка трудозатрат исполнит. (ч)	Оценка трудозатрат с учетом рисков (15%) (ч)	Общая оценка трудозатрат на задачу
Верстка по прототипам	Разработка верстки на основе Bootstrap 4 по разработанному ранее прототипам	Программист (back-end)	0	0,00	13,11
		Программист (front-end)	8	9,20	
		Тестировщик	2,4	2,76	
		Руководитель проекта	1	1,15	
Проектирование БД	Проектирование структуры БД с возможностью хранения прошлых версий данных, которые были изменены (версионирование)	Программист (back-end)	12	13,80	14,95
		Программист (front-end)	0	0,00	
		Тестировщик	3,6	4,14	
		Руководитель проекта	1	1,15	
Разработка БД	Разработка БД и хранения процедур	Программист (back-end)	16	18,40	23,92
		Программист (front-end)	0	0,00	
		Тестировщик	4,8	5,52	

Таблица 3 — Оценка задач специалистами с заложенными рисками

Всё это собирается в файл под названием «Детальная оценка задач» (позже его переименуем и дополним).

В таблице 3 можно увидеть уже заполненную таблицу. Где стоит 0, там участие специалиста не запланировано. Оценка тестировщика рассчиталась автоматически — взято 30% от суммы разработки, в данном примере это сумма «Программист (Backend)» и «Программист (frontend)». Этого времени должно хватить на тестирование и корректировки.

Замечу:

В указанном примере тестирование является

частью работы над задачей, поэтому оно входит в общую оценку трудозатрат на задачу, но могут быть отдельные задачи на тестирование, которые оценивают отдельно.

— **Шаг 7.** Далее нужно определить, сколько доступно временных ресурсов специалистов в проектной команде, закреплённой за проектом (см. таблицу 4).

Период работы: 04.02.2019 - 15.02.2019				
ФИО сотрудника (роль)	Доступно (ч)	Отпуск (ч)	Эквивалент % в часах	% загрузки по проекту
Сотрудник 1 (front-end)	64,00	16,00	80,00	100,00%
Сотрудник 2 (Тимлид)	0,00	32,00	32,00	40,00%
Сотрудник 3 (front-end)	80,00	0,00	80,00	100,00%
Сотрудник 4 (back-end)	64,00	0,00	64,00	80,00%
Сотрудник 5 (back-end)	40,00	8,00	48,00	60,00%

Таблица 4 — Проектная команда и доступное время специалистов

В таблице 3 показан конечный результат оценок. Далее опишу последовательность действий для их определения. Проекты можно разделить на:

— Проекты с **установленной клиентом датой** окончания работ — это когда заказчик устанавливает сроки, а исполнитель оценивает их исходя из оценок проекта.

— Проекты с **называемой исполнителем датой** окон-

чания работ. В этом случае нужно спрогнозировать общее время работ командой над задачами, конечно с учётом возможных рисков и тестирования. Здесь оценка пока приближительная, её можно будет скорректировать на дальнейших шагах.

Допустим, руководитель проекта уже знает дату окончания работ, определяется дата начала. Это время берётся за 100%, после чего рассчитывается, сколько это в часах. В примере на рисунке 7 период работ 2 недели, это 80 рабочих часов, если у специалистов будут отпуска или какие-то другие отгулы на этот период, учитываем их и получаем время (колонка «Доступно»), которое сотрудник должен отработать за указанный период.

Сделаю отступление, для подведения промежуточного итога, что уже должно быть собрано из документов:

- Бизнес-требования клиента.
- Функциональные требования.
- Техническое задание.
- Детализированный список задач.
- Сценарии использования (на их основе будут сделаны сценарии тестирования).
- Проектная команда (с указанием ролей, периодами участия в проекте и процент участия в проекте).
- График отпусков.

Далее руководителю проекта нужно понять, как распре-

делили задачи, все ли специалисты загружены и нет ли перегрузок. Проще говоря — посмотреть равномерность нагрузки. Ранее я писал про проектную команду и доступное время специалистов, приводил пример таблицы. Её нужно дополнить всего одной колонкой «Фактическая загрузка». В ней суммируются данные из колонки «Оценка трудозатрат с учётом рисков (15%)» в план-графике с группировкой по специалистам (см. таблицу 3). Полученный вариант можно увидеть в таблице 5.

Период работы: 04.02.2019 - 15.02.2019					
ФИО сотрудника (роль)	Фактическая загрузка (ч)	Доступно (ч)	Отпуск (ч)	Эквивалент % в часах	% загрузки по проекту
Сотрудник 1 (front-end)	64,00	64,00	16,00	80,00	100,00%
Сотрудник 2 (Тимлид)	0,00	0,00	32,00	32,00	40,00%
Сотрудник 3 (front-end)	80,00	80,00	0,00	80,00	100,00%
Сотрудник 4 (back-end)	64,00	64,00	0,00	64,00	80,00%
Сотрудник 5 (back-end)	40,00	40,00	8,00	48,00	60,00%

Таблица 5 — Проектная команда с фактической загрузкой

Это даст чёткую картину того, насколько загружен специалист, сколько свободного времени у сотрудника или он перегружен. Опираясь на эти данные, руководитель проекта сможет перераспределить задачи в этапе, или если говорить про весь проект, то принять решение об упрощении функционала, или привлечении дополнительных ресурсов, чтобы

успеть в сроки (в случае установленных заказчиком сроков).

Тут стоит сделать отступление и описать, что же такое дополнительные ресурсы и что ими может быть. Ресурсы в нашем случае — это время специалистов, которое нужно для реализации проекта. А вот этот ресурс можно получить так.

— **Привлечь дополнительных специалистов.** Хорошо, если они есть внутри компании. Если их нет, придётся нанимать, а это время. Но чаще всего такой возможности нет: нет бюджета.

— **Задействовать дополнительное время текущих специалистов на проекте** (нужно договориться о дополнительных часах работы в будни и выходные, но есть риск, что не все могут так работать и повышается риск выгорания специалистов, по этим причинам не сильно перегружайте специалистов) за дополнительную мотивацию.

Руководитель проекта с проектной командой оценили все задачи, прописали чего не хватает, скорректировали оценки, определили доступные ресурсы, теперь нужно прописать сроки работы над каждой задачей. При расчёте этих дат нужно учесть отпуска, время взятия в работу задач тимлидом и тестировщиком. Пример такого расчёта показан в таблице 6.

Роль в проекте	Время задержки (ч)
Тимлид	1
Тестировщик	1

Таблица 6 — Время задержки до взятия в работу

Нужно понимать, что Тимлид не может взять в работу задачу и проверить её по ряду причин:

- У него есть организационные обязанности.

- У него есть задачи по разработке кода (конечно, намного меньше, чем у разработчиков, он отвечает за ключевые механизмы, если говорить просто), а задачи появляются у него часто.

Вводим понятие **«время задержки»**, чтобы понять через сколько задача будет взята в работу.

У тестировщика ситуация похожая, у него задачи идут потоком, и он должен закончить те, которые у него уже в очереди. В его случае также определяемся со временем задержки перед взятием в работу. В приведённом в таблице 6 примере это один час у обоих типов специалистов, но оно может быть разным. Данные оценки стоит прорабатывать с командой, они зависят от проекта, условий работы и от самих специалистов.

Здесь нет ничего страшного, это не лишняя формальность, а реальность, которую нужно учесть, без этого проект

получит «просадку» по времени выполнения, а значит срыв сроков. Лучше, если специалисты закончат раньше работы над задачей и перейдут к следующей, чем не выполняют работы. А так проектная команда закончит всё вовремя или раньше, что покажет руководителя проекта с хорошей стороны, ведь он правильно работает с рисками и хорошо умеет прогнозировать.

Теперь ничего не мешает ввести в план-график две колонки «Начало» и «Окончание». Всё просто, дата начала первой задачи совпадает со стартом работ. Берём «Общая оценка трудозатрат на задачу» и «Время задержки» (нужно учесть, что не всегда нужен Тимлид или Тестировщик на задаче), отсчитываем от даты начала и получаем дату окончания работ по задаче, конечно с учётом выходных, праздников и отпусков.

Замечу:

У одного специалиста задачи могут пересекаться в рамках одного рабочего дня. Например, если текущая задача в последний день требует не 8 часов, а меньше, оставшееся время можно использовать для начала следующей задачи.

Это нужно учитывать при планировании.

Ранее мы работали с документом «Детальная оценка задач». Теперь его логично переименовать в «План-график проекта...» — вместо точек указывается название проекта. Далее привожу пример части плана-графика с датами и пе-

ресечениями, но без визуального отображения, то есть без закрашенных «прямоугольников» и временной шкалы (см. таблицу 7).

Наименование	Описание	Исполнитель	Оценка трудозатрат исполнит. (ч)	Оценка трудозатрат с учетом рисков (15%) (ч)	Общая оценка трудозатрат на задачу	Начало	Окончание
Верстка по прототипам	Разработка верстки на основе Bootstrap 4 по разработанным ранее прототипам	Программист (back-end)	0	0,00	13,11	23.01.19	24.01.19
		Программист (front-end)	8	9,20			
		Тестировщик	2,4	2,76			
		Руководитель проекта	1	1,15			
Проектирование БД	Проектирование структуры БД с возможностью хранения прошлых версий данных, которые были изменены (версионирование)	Программист (back-end)	12	13,80	14,95	24.01.19	25.01.19
		Программист (front-end)	0	0,00			
		Тестировщик	3,6	4,14			
		Руководитель проекта	1	1,15			
Разработка БД	Разработка БД и хранимх процедур	Программист (back-end)	16	18,40	23,92	28.01.19	30.01.19
		Программист (front-end)	0	0,00			
		Тестировщик	4,8	5,52			

Таблица 7 — План-график с датами начала и окончания работ

Теперь можно отложить периоды работы по задачам каждого специалиста на временной шкале, это даст более легкое восприятие информации. В таблице 8 привожу пример визуального отображения по одной задаче.

Исполнитель	Деление месяцев на недели								
	Январь				Февраль				
	1	2	3	4	1	2	3	4	
Сотрудник 1 (Тимлид)									
Сотрудник 4 (back-end)									
Сотрудник 4 (back-end)									
Сотрудник 3 (front-end)									
Сотрудник 4 (Тестировщик)									
Сотрудник 5 (UI/UX дизайнер)									

Таблица 8 — Визуальное отображение работ специалиста по задаче

В приведённом в таблице 8 примере можно увидеть разделение по неделям, но чаще всего используют разбивку по дням, которая даёт более чёткое понимание плана работ. Более подробно о визуальном отображении напишу далее в пункте 2.5.5.

2.5.3. ТРЁХТОЧЕЧНЫЙ МЕТОД ОЦЕНКИ ЗАДАЧ

Выше я писал про оценку времени, необходимого на решение задач. Но если есть желание получить более точную оценку или нужно дополнительно перестраховаться, при этом не очень хочется слепо доверять прогнозам членов проектной команды, стоит воспользоваться трёхточечным методом оценки.

Этот метод предназначен для более точного прогнозирования реального времени, которое потребуется на выполнение задачи или группы задач. Его смысл в том, что вместо одной оценки команда рассматривает сразу три возможных сценария: оптимистичный, наиболее вероятный и пессимистичный.

Трёхточечная оценка полезна не потому, что она выглядит «математически красиво». Её главная ценность в том, что она заставляет исполнителя думать не только об идеальном сценарии, но и о реальности.

При использовании трёхточечного метода оценки необходимо определить для каждой задачи три временных оценки на базе предыдущего опыта, экспертного мнения или наиболее вероятных предположений.

Для более точного расчёта продолжительности работ над задачей данный метод оценки можно представить в виде

формул, где:

— **О** = оптимистичная оценка, то есть самый благоприятный прогноз;

— **С** = реалистичная / средняя оценка, то есть наиболее вероятный прогноз;

— **П** = пессимистичная оценка, то есть самый неблагоприятный прогноз;

— **Е** = итоговый прогноз;

— **4 и 6** в первой формуле представляют стандартный метод, придающий больший вес наиболее реалистичному значению;

— **3** во второй формуле используется для расчёта простого среднего значения.

— **Оптимистичная оценка** показывает, сколько времени потребуется, если всё пойдёт идеально: не будет задержек, уточнений, ошибок, зависимостей от других людей и внешних проблем.

— **Реалистичная оценка** показывает, сколько времени задача обычно занимает в нормальных рабочих условиях: с обычными вопросами, согласованиями, переключениями и небольшими уточнениями.

— **Пессимистичная оценка** показывает, что произойдёт, если появятся сложности: дополнительные требования, баги, задержки согласований, зависимость от других специалистов, необходимость исследования или переделки.

Значения **О**, **С** и **П** определяются экспертно — в часах,

дней или денежных средствах — в ходе обсуждения с проектной командой.

Для этого участникам можно задавать одинаковые вопросы:

— Сколько времени займёт задача или проект, если всё пойдёт хорошо, без рисков и проблем?

— Сколько времени задача займёт при обычном рабочем сценарии?

— Каким может быть самый негативный сценарий и сколько времени или трудозатрат он потребует?

— Что может помешать выполнению задачи?

— Какие зависимости, уточнения, согласования или проверки могут увеличить срок?

— Есть ли у нас опыт выполнения похожих задач в прошлых проектах?

После этого полученные значения О, С и П подставляются в одну из двух формул: вариант с весом наиболее вероятной оценки показан в формуле 1, вариант простого среднего — в формуле 2.

$$E = (O + 4C + P) / 6$$

Формула 1 — Вариант расчёта прогноза №1

Эта формула часто используется в PERT-оценке. В ней наиболее вероятная оценка получает больший вес, потому

что чаще всего именно она ближе к реальному сценарию выполнения задачи.

$$E = (O + C + П) / 3$$

Формула 2 — Вариант расчёта прогноза №2

Вторая формула известна как треугольное распределение. Основное отличие состоит в том, что в этом методе не придаётся больший вес наиболее вероятному значению. Все три оценки учитываются равномерно, а результат расчёта даёт усреднённую оценку.

Такой расчёт позволяет учесть возможные позитивные и негативные сценарии, оценить их влияние и получить более реалистичный прогноз. Да, он занимает чуть больше времени, чем обычная оценка «на глаз», но это время часто окупается уже на этапе исполнения проекта.

PERT не усложняет оценку ради усложнения. Он легализует сомнения исполнителя и делает их частью плана. Вместо того чтобы скрывать неопределённость за одной красивой цифрой, команда честно показывает диапазон возможных вариантов.

Например, исполнитель может сказать **«Если всё пойдёт хорошо, задача займёт 6 часов. Скорее всего — 10 часов. Если появятся сложности с интеграцией, может занять 18 часов»**, тогда такая оценка гораздо полезнее для руководителя проекта, чем простое **«Наверное, сделаю за**

день».

Трёхточечная оценка помогает не только посчитать срок, но и увидеть, где в задаче находится неопределённость. Если разница между оптимистичной и пессимистичной оценкой слишком большая, это сигнал для руководителя проекта: задачу нужно уточнить, разбить на части, провести дополнительное исследование или заложить резерв.

Управленческий принцип:

План без буферов — это не план, а пожелание.

Буфер — это не признак слабого планирования. Это профессиональный учёт неопределённости и управляемых рисков проекта. Если в проекте нет резервов, любая задержка, уточнение или ошибка сразу начнёт разрушать график.

Важно помнить:

Буфер не должен превращаться в скрытый запас «на всякий случай», который никто не контролирует. Его нужно использовать осознанно — для покрытия реальных рисков, уточнений, задержек и непредвиденных обстоятельств.

Трёхточечный метод особенно полезен, когда:

- задача новая для команды;
- есть техническая неопределённость;
- результат зависит от внешних участников;
- есть риск изменения требований;
- исполнитель сомневается в одной точной оценке;
- задача находится на критическом пути;

— ошибка в оценке может повлиять на срок всего проекта.

Запомните:

— одна оценка часто скрывает неопределённость;

— три оценки помогают увидеть диапазон возможных сценариев;

— большая разница между оптимистичной и пессимистичной оценкой означает высокий уровень неопределённости;

— PERT помогает встроить сомнения исполнителя в план, а не игнорировать их;

— буферы нужны не для красоты, а для управляемости проекта.

Столько работы уже проделано: мы начали переходить от отдельных задач к более реалистичным срокам. Осталось совсем немного, но следующая часть не менее интересная, чем планирование ресурсов и оценка. Дальше нужно будет собрать задачи, оценки, зависимости и доступность команды в понятный план-график.

2.5.4. РЕДЛАЙН: ВНУТРЕННЯЯ КРАСНАЯ ЛИНИЯ СРОКА

Сейчас хочу поговорить об одном небольшом, но очень полезном управленческом инструменте. На первый взгляд он кажется простым: всего лишь поставить внутренний срок раньше внешнего срока сдачи. Но на практике именно такие простые вещи часто спасают проект от аврала, нервов и неприятных разговоров с заказчиком.

Речь пойдёт о редлайне.

Все знают слово «дедлайн». Все понимают, что финальный срок нарушать нельзя. Но проблема в проектах обычно возникает не в сам день сдачи, а раньше — когда команда слишком поздно понимает, что результат ещё сырой, тестирование не закончено, документы не подготовлены, а времени на спокойные исправления уже нет.

Именно для этого и нужен редлайн.

Редлайн от англ. **redline** — «красная линия» — это внутренний контрольный срок, к которому работа должна быть готова для проверки, исправлений, тестирования, согласований и спокойной финальной сдачи.

Проще говоря, редлайн — это момент, когда команда должна перестать **«просто делать»** и перейти к проверке готовности результата.

Внешний срок сдачи обычно видит заказчик, руководи-

тель или спонсор. Редлайн же нужен внутри команды. Он показывает, когда работа должна быть готова настолько, чтобы у руководителя проекта осталось время проверить результат, найти ошибки, организовать исправления и подготовить спокойную сдачу.

Если внешний срок стоит на пятницу, 18:00, грамотный руководитель проекта не планирует завершение работ на пятницу, 17:59. Он ставит внутренний редлайн раньше: например, на среду вечер или четверг утро. Оставшееся время используется не для героического добивания задач в последний момент, а для проверки результата, исправления ошибок, согласования и подготовки к сдаче.

Это не бюрократия и не перестраховка. Это нормальный управленческий инструмент.

Если вы запланировали завершение работы ровно к финальному сроку, вы уже опоздали управленчески. У вас не осталось пространства для ошибки.

Редлайн нужен не для того, чтобы давить на людей раньше. Его цель — создать управляемый запас времени между внутренней готовностью и внешним обязательством. Собрал список то, в чём он вам будет помогать:

- заранее увидеть реальное состояние проекта;
- защитить проект от форс-мажоров;
- оставить время на исправления, тестирование и согласования;
- улучшить качество результата;

— снизить стресс перед сдачей.

Заострю ваше внимание на том, что редлайн и буфер связаны, но это не одно и то же:

— **Буфер** — это запас времени, который защищает проект от неопределённости.

— **Редлайн** — это контрольная точка перед этим запасом, говорящая, что пора начинать использовать этот запас осознанно.

Связь между редлайном, буфером и дедлайном показана на рисунке 3.



Рисунок 3 — Редлайн, буфер и дедлайн в управлении сроками

Например:

— внешний срок сдачи: пятница, 18:00;

— редлайн: среда, 18:00;

— буфер: четверг и пятница до сдачи.

Редлайн показывает момент, когда работа должна перейти из режима «делаем» в режим «проверяем, исправляем, сдаём».

Как определить редлайн? Универсального процента нет. Но для начинающего РП можно использовать простое правило: редлайн ставится на 10—30% раньше финального срока, в зависимости от сложности, неопределённости и критичности результата.

Например:

— если задача маленькая и понятная, достаточно 10% запаса;

— если задача связана с тестированием, согласованием или несколькими исполнителями, лучше заложить около 20%;

— если проект сложный, есть внешний заказчик, интеграции, подрядчики или высокая цена ошибки, нужен запас ближе к 30%.

Важно:

Редлайн нельзя ставить «из воздуха». Он должен быть связан с реальными действиями после него.

После редлайна должно остаться время на:

— тестирование;

— исправление ошибок;

— ревью результата;

— согласование с заказчиком или руководством;

- подготовку документов;
- упаковку релиза;
- проверку доступов, инструкций и сопроводительных материалов;
- финальную коммуникацию со стейкхолдерами.

Если после редлайна ничего не происходит, значит это не редлайн, а просто ещё одна дата в календаре, которую команда быстро перестанет уважать .

Например:

Команда должна показать заказчику готовый функционал в пятницу в 15:00.

Плохой план:

- разработка заканчивается в пятницу в 13:00;
- тестировщик проверяет «на бегу»;
- баги чинятся за час до демонстрации;
- РП параллельно пишет письмо заказчику;
- команда нервничает;
- на демонстрации всплывает ошибка.

Формально команда «почти успела». Управленчески — нет.

Хороший план:

- редлайн на разработку: среда, 18:00;
- четверг: тестирование и исправления;
- четверг вечер: внутренняя демонстрация РП и техли-

ду;

— пятница утро: финальная проверка и подготовка показа;

— пятница, 15:00: спокойная демонстрация заказчику.

Разница не в том, что команда стала работать медленнее. Разница в том, что руководитель проекта перестал делать вид, будто между «закончить задачу» и «сдать результат» нет работы.

Главная ошибка — превращать редлайн в скрытый финальный срок.

Если РП говорит команде: «**Сдать нужно в среду**», хотя реальная сдача в пятницу, но не объясняет зачем, команда быстро почувствует манипуляцию. Один раз это может сработать. Потом люди поймут, что настоящий срок всё равно позже, и редлайн теряет силу.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.