

Наталья Легеза

КАК ПРОГРАММИСТЫ НАЛЕВО ХОДИЛИ



СОДЕРЖИТ
НЕЦЕНЗУРНУЮ
БРАНЬ

18+

Бизнес-советы предпринимателям

Наталья Легеза

**Как программисты налево ходили.
Бизнес-советы предпринимателям**

«Автор»

2024

Легеза Н. И.

Как программисты налево ходили. Бизнес-советы
предпринимателям / Н. И. Легеза — «Автор», 2024

МИФ ИЛИ РЕАЛЬНОСТЬ ОБ ИЗМЕНАХ ПРОГРАММИСТОВ СВОИМ РАБОТОДАТЕЛЯМ. А можно ли это считать изменой? Ведь каждый вправе в свободное от работы время делать то, что он хочет. У нас же не рабовладельческий строй с разделением на Хозяев и Рабов. А значит, и подработку на стороне вправе назвать Хобби. Может кто-то картины собирает, а кто-то коды программ коллекционирует, чтобы не умереть от скуки на рутинной работе. А что на этот счет думают владельцы бизнесов? Думаю, что не думают от слова «**ВООБЩЕ**». А чего его думать? Вон в уголочке программист свой сидит в джинсах и в свитерке, прямо так уютно, по-домашнему. Работает себе, да и ладно, главное, что при деле, и всегда если что можно послать свой ноутбук отвезти в ремонт. Штатная единица на то она и штатная, в полном распоряжении на 8 часов, пользуйся по всем рабочим вопросам, только зарплату плати вовремя. У автора на этот счет конечно же своё мнение, кто ж как не она знает все секреты программистов, которые они скрывают от любопытных глаз.

© Легеза Н. И., 2024

© Автор, 2024

Содержание

Вступление	5
Глава 1. Программист – свой или чужой?	7
Глава 2. Порассуждаем о размерах, компаний конечно	11
Глава 3. Программисты не вечны, и вы тоже.	17
Конец ознакомительного фрагмента.	19

Наталья Легеза

Как программисты налево ходили. Бизнес-советы предпринимателям

Вступление

Никогда не считай себя не таким, каким тебя не считают другие, и тогда другие не сочтут тебя не таким, каким ты хотел бы им казаться.

«Алиса в стране чудес» Льюис Кэрролл

В прошлом веке, в далёком 1997 году, я впервые познакомилась с программой 1С. Это была 1С:6 версия для бухгалтерии. Обучать премудростям работы в данном программном обеспечении в Корпорации «Борис» (моё первое место работы) поручили системному администратору Константину по вечерам после работы. Когда офисные работники расходились, после 18:00, мы с Константином учились как пользоваться программой на свободном компьютере. В университете нас, к сожалению, на уроках информатики садили по три человека на деревянной скамье за один компьютер, и максимум что мы усвоили так это то, как включить и выключить компьютер, ну и пару раз стукнули по клавишам. Соответственно, находясь уже в возрасте 20 лет, компьютерной грамотности у меня было аж целых ноль с хвостиком. Уроки в Корпорации не прошли даром, основы были заложены правильно, а значит дверь в 1С была открыта.

«А вы знаете почему 1С, называется именно «1С»? Когда-то очень давно в 1995 году основатель компании Борис Нуралиев заметил, что программа выполняет запрос пользователя на интерфейсе за 1 секунду. Отсюда и пошло название «1С» – 1 секунда»

Откровенно хочу сказать о том, что любовь к 1С у меня была с первого взгляда, и за всё это время я своим чувствам не изменила, она для меня верный друг и товарищ, правдивый и честный, лекарство от лжи и обмана нечестных людей.

После 1С:6 версии пришла 1С:7.7, посложнее, поинтереснее. Новая автоматизация, новые знания, и испытания конечно же тоже. В то время я уже работала главным бухгалтером в Корпорации, сместив прошлого спившегося шампанским не столь ценного кадра Зинаиду Ивановну.

На новую версию 1С:7.7 мы перешли, что и стало моим первым опытом управления проектом по автоматизации учёта компании бензиновой отрасли, с миллионными оборотами.

Первые знакомства с программистами 1С в Крыму, новый опыт коммуникаций с особенной кастой разработчиков.

С тех пор прошло уже 25 лет, я уже работаю с 1С:8.3, а версию 1С:7.7 называю динозавром, и не берусь управлять проектами с участием этой «старушки».

Зачем написала эту книгу?

Ну, во-первых, книги я писать люблю.

Ну, а во-вторых, хочу поделиться своим опытом, искренне, от всей души. Опыт оплачен временем, деньгами, слезами, нервами, ссорами, расставаниями. Всё как в хорошем блокбастере, с бурей эмоций и минусом на кредитной карте.

Ещё признаюсь в любви всем тем людям, которые сейчас рядом со мной, бок о бок, решают вместе со мной задачи, разбираются в проблемах, выслушивают.

Все мои суперзамечательные программисты 1С, бухгалтера, мои администраторы проектов, и мои клиенты, Я ВАС ИСКРЕННО ЛЮБЛЮ.

В любви призналась, поехали дальше.

Будем разбираться на страницах этой книги, почему программисты изменяют своим работодателям, ходят налево, смотрят по сторонам и ни в какую не желают годами сидеть на одном месте.

Если вы думаете, что всё упирается в деньги, то ошибаетесь (моё субъективное мнение), так как практика показывает, что не всегда материальная составляющая и есть та единственная причина, когда программист соглашается на другие проекты. Ситуация комплексная, поэтому сразу же предупрежу вас, что не ищите в этой книге, как сделать так, чтобы программист работал только на вас. «Не пасите котов», то есть не пытайтесь взять контроль управления разработчиком 100% в свои руки, это однозначно не понравится сотруднику и приведёт только к скорейшему расставанию. Да и кому понравится ежедневный контроль и бдящее око сутками напролёт, вам лично такое вряд ли понравилось бы.

Попробуем на протяжении всей этой книги порассуждать об ИТ сфере, о взаимодействии с разработчиками, ведении ИТ проектов. Немного скриптов вам в ленту добавлю, сказку расскажу, да и ещё инструкцию для составления партнёрского соглашения добавлю, как вишенку на торте.

Если вы хотите быть успешными в области ИТ сферы, быть результативными, мои рассуждения и выводы вам в этом частично помогут.

Почему частично?

Да потому, что в любой области, претендуя на экспертность надо книг прочитать не меньше 100, и поработать в данной отрасли не менее 5 лет, и только потом вы начнёте лучше понимать прежде всего себя, ваше поведение, ваши решения, а потом и до остальных окружающих специалистов дотянетесь.

Любой просчёт в бизнесе может обернуться или потерей денег, или потерей сотрудников, а может и всё вместе взятое.

Не большая сумма не страшна, но значительное вложение денежных средств в проект, который не воплотился в бизнес процесс и не принёс дополнительного заработка, или не с оптимизировал расходы, помимо нервного истощения, приведёт к истощению ресурса компании, потере ликвидности в деньгах и образует финансовую яму.

О чём ещё книга?

О деньгах конечно.

И о том, как не потерять кровно заработанные, и не только когда в стране форс-мажорные обстоятельства, а при любой политической ситуации.

А не потеряв ресурс, и добившись всё-таки результата в ИТ проекте, заработать больше.

В книги приведены живые достоверные кейсы, из категории «неудавшиеся проекты», все имена героев в книге изменены, за исключением моего конечно.

Рассказывать об успехах и победах конечно легче и приятнее, но кто-то должен показать и обратную сторону бизнеса, дабы предотвратить «удары граблями» (из-за того, что на них наступили и не один раз) последующих поколений подрастающих предпринимателей.

Помните, что настоящие бизнесмены – мечтатели и фантазёры с богатым воображением.

Ну а как иначе?

Без этого бизнес не построишь.

Кстати если заметите, что в книге много фраз и выражений из книги Льюиса Кэрролла «Алиса в стране чудес», не удивляйтесь, моё путешествие в мир бизнеса практически ничем не отличается от истории, изложенной в книге, за исключением только других действующих лиц, отличающихся обстоятельств и времени событий. В остальном всё равно тоже самое, падение в кроличью нору, и пошло-поехало.

Глава 1. Программист – свой или чужой?

А где я могу найти кого-нибудь нормального?

– Нигде, – ответил Кот, – нормальных не бывает. Ведь все такие разные и непохожие.

И это, по-моему, нормально.

«Алиса в стране чудес» Льюис Кэрролл

Из практической жизни.

Задача программисту 1С:8.3.: «Закрасить перламутром в квадрате и в зонтике».

Первая мысль, которая вполне вероятно приходит в вашу голову, когда вы читаете название данной главы, наверняка о том, что я собралась рекламировать свою компанию и хвалить своих программистов. Предполагаете, что сейчас начнутся с моей стороны пиар лозунги: «гоните взащей штатных программистов» и берите моих, они красивее, моложе и умнее.

Нет не угадали. Если бы вы прошли мой путь 20 лет взаимодействия с штатными разработчиками и ещё 7 лет с фрилансом, то нам было бы о чём подискутировать. Хотите верьте, хотите нет, но руководя и внедряя более сотни проектов по автоматизации бизнеса, я сняла розовые очки только 3 года назад, когда объективная реальность отрезвила мой ум, и стало явно и очевидно, «кто есть на самом деле программист в штате».

Парадоксально, но факт, мне это было не понятно более 20 лет.

Конечно же я слышала, и читала, что IT стартапы горят как мотыльки, летящие к огню, но считала, или наивно полагала, что причина в миллионе факторов, такие как условия на рынке, недостаток финансирования, невостребованный продукт и т.д.

Я ОШИБАЛАСЬ.

И считаю на сегодня на 100%, что во главе угла IT компании должен находиться НЕ ПРОГРАММИСТ, а ЛИДЕР, у которого есть ВИДЕНИЕ.

ДА! Именно ВИДЕНИЕ, того что нужно делать.

До тех пор, пока компании будут сажать на трон и одевать корону программисту, никакого прогресса в воплощаемых проектах не будет.

Огромное заблуждение большинства людей состоит в том, что они думают, ЧТО ПРОГРАММИСТ ЗНАЕТ, ЧТО ДЕЛАТЬ!

Нет! Верным суждением будет: «ПРОГРАММИСТ БУДЕТ ЗНАТЬ, ЧТО и КАК ДЕЛАТЬ, КОГДА ЕМУ УКАЖУТ, ЧТО ДЕЛАТЬ И ПОКАЖУТ НАПРАВЛЕНИЕ ЛОГИЧЕСКОЙ МЫСЛИ».

Прошу обязательно придать значение двум словам «ЛОГИЧЕСКОЙ МЫСЛИ».

Т.е. логика развития событий должно быть строго последовательной, как маршрут на карте выстроенный GPS навигатором.

Конечно же маршрут может быть скорректирован со стороны программиста, на то он и опытный «пилот», тех кто «не в зуб ногой» в расчёт не берём.

Советы принимаются, как обойти те или иные препятствия, но направления господа, да, да, именно направление и точка конечной остановки должна быть ясна и понятна обоим сторонам.

Это вам не сказка «иди туда не знаю куда, принеси то, не знаю, что». За чем пошлёте, с тем и вернутся, а может и не вернуться, тут уж по обстоятельствам.

Лучше не беритесь руководить проектами, если в голове у вас поселилось ННН.

Не знаете цели проекта,

Не в курсе, какой результат ожидается,

Не собираетесь погружаться в детали.

ННН чревато перерасти в ППП

Потеря денег

Потеря репутации

Пиндец («н» можете заменить на «з», разрешаю)

А теперь попробуем все вместе догадаться, почему при решении IT задач лучше обращаться к профессиональным компаниям, т.е. компания должна специализироваться в этой отрасли.

Как определить профессиональная компания или нет, попробуем разобраться далее по книге.

А сейчас попробуем сделать выбор куда нам бежать и к кому обращаться.

Хочу акцентировать внимание на вопросе того, что абсолютно не имеет значение то, какого размера ваша компания и сколько в ней человек. Объясню почему это не важно. По устоявшемуся мнению, в бизнес кругах, если компания крупная, то обязательно программисты должны присутствовать в штате и руководить ими будет директор IT департамента, который по большей части является эдаким «человек-железо» (не путать с «IRON MAN»), т. е. тот, кто разбирается в серверах и безопасности.

Почему?

Да потому, что неприкосновенность данных в компании и бесперебойная работа железа (в виде серверов и их составляющих) сегодня занимает более важное место, чем новые программные разработки. И это абсолютно правильно, т.к. вся компания парализуется на неопределённое время, если с IT инфраструктурой произойдёт непредсказуемая ситуация в виде сбоя работы инфраструктуры, ну а следом накроет коллапс.

Теперь, помыслив логически, мы приходим к выводу, что, если программистами будет руководить человек, который имеет базовые знания написания кода, и может даже сам что-то написать умненькое, по факту, в проекты вникать не будет. Разбираться в задачах, которые ставят операционные менеджеры разработчикам остаётся в зоне влияния самих же разработчиков.

Представим себе такую картину, что в какой-то комнате в полной изоляции сидят несколько людей, чем они там занимаются, конечно, никто особо не разбирается, так как допуск в комнату посторонним запрещён. Через маленькое окошко в эту самую комнату подаются написанные задания, и оттуда через какое-то время (через какое никому точно неизвестно, сроки передвигаются, задач много, нервов на всех не хватает) выдаётся некий результат.

В момент возмущения выданными данными, приходит ответ, возможный вариант ответа на листике бумаги от руки написанный, следующий:

«Устраивает, не устраивает, а что вы хотели, как написали в письме, так и сделали. Написано же «семь зелёных клеток с точками, вот и получите. Что? Хотели точки над зелёными клетками и клетки в один ряд, а не на друг друге? Ну извините, мы тут заняты, нам тут домысливать некогда. Пишите задачу на доработку».

И вот теперь вы только начинаете понимать и потихоньку догонять, что на самом деле в отделе программистов происходит.

Думаете они там в компании сидят, работают, что аж дым из ушей валит, вон гляди всё кофе опять выпили.

Открою вам небольшой секрет, вам знакомо выражение: «солдат спит, а служба идёт»?

Так вот и у программистов, которые устраиваются в штаты компаний, чтобы иметь базовый оклад, запись в трудовой книжке (а то как-то 18 лет фрилансера, в резюме не очень смотрится), и спокойно себе искать проекты на стороне, то же самое.

Я очень долго смеялась, когда один дядечка с очень большой компании рассказывал о таких методах работы контроля программистов, как ежечасно делать снимок рабочего экрана, чтобы понимать, что бедолага сейчас делает и чем занят. Ну фотографируйте экран, а этот самый товарищ всю ночь не спал, проект на стороне «пилил», сидит пятую кружку кофе пьёт и в абсолютно не ресурсном состоянии.

Ну а что?

Он думает так: «мышкой» по экрану повожу, что-то напишу, потом опять перепишу, потом скрипт обновлю, потом на отладку поставлю, и опять по кругу. Там глядишь и день прошёл, и зарплата скоро. Сиди себе кофе прихлёбывай, да сахарку побольше ложи. А кто померяет сколько наработал? Вон глядишь начальник мой, тоже к машине побежал, там у него ноутбук личный запрятан, что-то срочно решить другому клиенту надо. Клиентами делится, не жадничает, только с часа процент отдавай. Так что город засыпает, а мафия просыпается. И наоборот.

Наверное думаете, что я выдумываю, и это только некоторые недобросовестные разработчики так делают.

А вы посидите и подумайте, только очень хорошо, логически выстройте цепочку.

В большинстве своём разработчики – это мужчины. Устройство в компанию сродни определённому брачному контракту, в котором базовые обязательства соблюдены, а вот в свободное время, это ещё надо подумать кто кому и что должен.

Каждый день одно и то же, разнообразием не пахнет, одна и та же компания, одни и те же надоевшие лица, по типу с претензиями «опять программа не работает, потому что шнур уборщица задела, или того хуже проливающие чай и кофе на клавиатуры, но при этом утверждающие что ничего подобного в их жизни никогда не случилось.

Скучно, батенька, скучно тут с вами. А вот там на стороне – ОПЫТ, ГОСПОДА. Разнообразие, за которое, между прочим, ещё и деньги предлагают, и что самое важное даже похвалить могут, а не дармоедом обзывают, и в отпуск только в феврале и ноябре отпускают, когда все остальные летом и на новогодние праздники, из которых только к середине мая возвращаются.

Поэтому, извините конечно, но свобода движений, новые контакты, новые вызовы ни в какое сравнение с вашей серой будничной унылой жизнью даже рядом не стояли. Поэтому летят программисты к новому и неизвестному. А на работе что, могу и с открытыми глазами поспать пока компьютер перезагружается. А вы попробуйте словить на бездействии, да ничего у вас не получится, это вам не яму копать, и не болванку на станке клепать, по норме 2 болванки без брака в час. В 1С таких болванок не настрогаешь, тут знаете индивидуальный творческий подход нужен.

Рассказов про то, почему так долго делали задачу, я в своё время наслушалась очень много, и до сих пор выслушиваю сердобольные истории.

На элементарный вопрос, почему отчёт, который можно за 8 часов собрать, а не три недели ждать мне ответили таким образом: «А вы знаете сколько мне над этим пришлось думать?»

Знаю батенька, знаю, не фига ты не думал, дурака валял, и другими делами занимался, поэтому иди-ка ты подбру-поздорову, к другим олухам, кто в твои сказки для Панаски поверит. Я в детективные истории про «долго думать» не верю никак, потому, что тому разработчику, которому действительно нужно задачу решить, он и посоветуется и техническое решение выдаст как можно быстрее. И всё потому, что он хочет и задачу закрыть, и денег получить, именно это его и вдохновляет. А прохиндеев на рынке хватает, не успеваешь избавляться, как вон уже новый на горизонте.

Практический совет, как спокойно без нервов избавиться от бестолкового программиста.

Никто не застрахован, и я тоже, получить в команду разработчика не умного, но скандального, от которого могут случиться неприятности. И работа не будет сделана, и клиент скандалит с одной стороны, а с другой стороны разработчик требует денег.

Совет первый: не скрывайте ситуацию от клиента, если есть возможность заменить программиста сделайте это немедленно, не тяните время.

Совет второй: при первых признаках, что программист с «заскоком», блокируйте доступ, меняйте пароли, отстраняйте от работы. Не медлите. Как распознать «заскоки», ну вы же не вчера, я надеюсь, бизнес открыли. Попросите совета и рекомендацию другого программиста, чтобы он адекватно проанализировал ситуацию, если вы сами не очень, пока что, разбираетесь.

Совет третий: включите сразу другого разработчика, с целью, чтобы убедиться, что код в базах не зашифрован, и необходимые модули присутствуют. Это можно проверить, подняв прошлые бэкапы (архивированные копии баз).

Совет четвертый: если программист начинает требовать от вас деньги за работу, которую он не выполнил в первую неделю возникшего конфликта, не нужно обострять ситуацию до критического момента. Любые споры разгораясь всё больше, если в них «подбрасывать дров», первые две недели от момента возникновения инцидента нужно всеми возможными усилиями сглаживать острые углы. С клиентом вы проблему уладите, но вот с разработчиком дело будет сложнее. Он потратил время, и его претензия будет заключаться в шантаже и вымогательстве с целью забрать денег в наглую и по-хамски. В данном случае не пройдет просто заблокировать номер, это может вызвать неадекватную редакцию (даже не возьмусь предсказывать, что может произойти). Нужно проводить разъяснительную работу, в спокойном тоне, не раздражаясь и не выливая негативные эмоции. Время будет играть на вашу пользу, если вы, подкрепляясь фактами, будете обосновывать ваш отказ оплаты, прежде всего отсутствием результата. Прошу понять, что причина «отсутствие результата» и «ты бестолковый» – это две большие разницы. Почему результат не случился, надо проанализировать объективно и сделать совместные выводы. Просто «отморозиться» и «абстрагироваться» от проблем не пройдет. Спокойно относитесь к угрозам, с посыланием на вашу голову всех проверок, аудитов, и кары небесной. Если бы этот разработчик был действительно такой серьезный, сидел бы он в рядовых программистах на шабашках? Да нет конечно, уже бы сам своей фирмой руководил, а не пыль в глаза пускал.

Какие выводы можно сделать из этой главы?

Своего родного разработчика оставить, вон в углу в стареньких джинсах сидит копаются, или гнать его вон и нанимать при случае экспертов?

Конечно же вам решать, вам ведь виднее. Вы же своим бизнесом руководите, а не я, вы и думайте головой.

Глава 2. Порассуждаем о размерах, компаний конечно

Подумать только, что из-за какой-то вещи можно так уменьшиться, что превратиться в ничто.

«Алиса в стране чудес» Льюис Кэрролл

Любая из больших компаний – это колосс на глиняных ногах. Можете со мной не согласиться, но я вас уверяю, что любой из бизнесов можно разрушить, всего лишь создав информационное поле о его неплатежеспособности, подкреплённой отсутствием своевременных платежей по счетам поставщиков. Пару тройку месяцев и любой гигант завалится, и на обломках ещё долго будут бродить коллекторы, кредиторы и банки подбирая крошки среди дымящихся развалин.

Про банки и истории с коррумпированными банковскими служащими можно писать много, и о том, как бизнес банкротится сознательно с целью не вернуть кредиты, но книга не об этом, поэтому оставим эти рассуждения для следующих изданий.

Чем больше бизнес, тем больше в нём размытости, неопределённости, неуверенности в декларируемой прибыли. Собственники, вырастив компанию до оборотов сотню миллионов долларов в год, могут теперь только ориентироваться на сумму денег, которую они могут выручить в случае продажи бизнеса. Другим показателям, из-за потери контуров операционной деятельности, доверие низкое, по причине отрицания (внутреннего конечно) о том, что такая машина может нести и убытки тоже.

Такие компании просто клондайк для IT сектора. Бюджеты подтверждаются, не без проблем конечно, но всё же в приличных размерах. Обычным делом в таких корпорациях считается назначение при старте проекта одного ответственного, в середине проекта – следующего, ну а дальше конца и края конечно действующим лицам не видно. Там смотришь, уже и разработчики сменились, и фамилию первого ответственного в компании никто уже и не вспомнит.

Солдат спит, а служба идёт. Код пишется, все трудятся в поте лица, бюджет осваивается в полном объёме.

Ну а если проект не удался, так кто виноват?

Ну конечно же тот, кто был первым ответственным за проект.

А где он делся?

Уволился, и след его простыл.

Почему уволился?

Так вот в записке перед уходом написал: «Не вижу даты окончания проекта. Прошу в моём увольнении никого не винить».

Компания большая, время идёт, и паровоз мчится дальше, таща за собой вагоны и маленькие тележки ещё таких же проектов в разных областях. Сумму затрат старательная бухгалтерия спишет на убытки, и через год уже никто ничего не вспомнит, всё поросло забвением и мхом.

Нет, конечно же автор не хочет сказать, что это происходит во всех компаниях. Где-то конечно результат, какой-никакой получается. Пусть плохонький, на костыликах, но он есть.

Сложно измерить процент не сложившихся проектов на бизнес просторах за последние 30 лет независимости страны, да и кто же нам в этом сознается. Расписаться в неудачах, сложно, гораздо проще задекларировать успехи.

Если прикинуть навскидку, по статистике: из 100% IT стартапов – 99% прогорает, а 1% выживает.

Сколько из 1% выживет ещё через 5 лет? В такой же пропорции.

Исходя из этой логики, такой же и процент выполнения IT проектов в компаниях.

И опять же, а что можно считать результатом? Это вам знаете ли не свитер связать из бабушкиной пряжи, где объект более-менее понятен в своём масштабе, да и материал ощутим в своём качестве и количестве.

В больших компаниях есть определённая стратегия, которую используют недалёковидные менеджеры, цель которых как можно больше выслужиться перед начальством.

А как это сделать если особыми навыками и компетенциями не обладаешь?

Всё очень просто.

Для начала инициируется IT проект, более-менее приличных масштабов, чтобы в нем было задействовано как минимум пару тройку отделов компании, не считая самих разработчиков. Определяются контуры, и предварительные зарисовки, что нужно получить по итогу автоматизации. Далее, эти самые ушлые менеджеры, которые несомненно вызовутся руководить данным проектом, начинают приглашать представителей сторонних IT компаний, и предлагать им осуществить нарисованное на бумаге в жизнь. Начинается бурная деятельность: дискуссии, встречи, чаепитие, кофее питье, печенье дешёвенькие, вопросы, ответы, предложения, согласование бюджета. И так шаг за шагом вырисовывается план проекта, который становится более-менее понятен, потому, что в его разработке приняли участие IT компании, обладающие компетенцией в поставленной задаче. Финалом таких встреч является отправка со стороны потенциального исполнителя (читайте: «жертва обмана») коммерческого предложения со стоимостью данной разработки.

Что происходит дальше для IT компании, потенциального подрядчика?

Наверное, вы уже догадались.

ТИШИНА и МОЛЧАНИЕ.

На попытки добиться ответа, шаблон письма от менеджера всегда наготове:

«На согласовании у руководства».

Такое согласование не закончится никогда.

Почему?

Да потому, что в это время, наш герой (тот самый менеджер, про инвестировавший в дешёвые кондитерские изделия для встреч), побеседовав со своими штатными разработчиками, рисует свои цифры бюджета, которые будут сделаны сотрудниками компании, включая премиальный фонд (за экономию).

И вот всё это «детище», в виде презентации, торжественно несётся на несомненно важное совещание (других в больших компаниях не бывает), на котором показывают две картинки затрат: своими силами и чужими руками.

Вот вы лично, например, если пойдёте в магазин и увидите там две бутылки шампуня внешним видом ничем не отличающиеся, но разницей в цене 30% (учтите факт, что вы этим шампунем не будете лично пользоваться), какой выберете?

Правильный ответ: тот, от которого денег на вашей платёжной карте останется больше.

Тем более если ваш менеджер компании, всё изучил, тридцать ночей не спал, продумал всё до мелочей (то, что касается личного вознаграждения, а не проекта, если вы так подумали).

Всё! Проект утверждён, побежали выполнять.

То, что результат будет вдвое хуже, это уже по итогу никого не волнует.

Это как результат операции хирурга: уже никто не сможет сравнить качество выполненной работы, кроме другого квалифицированного специалиста, так как больной думает, что как сделали, так и должно быть.

Я с подобными компаниями на практике сталкивалась (обойдёмся без имён и названий, тем более, что часть из них уже канули в лету), и конечно же попала в ловушку доверия, но сделав выводы, пришла к правильному решению:

«Ни ногой к разработке технического задания, до тех пор, пока не будет понятно, кто за него заплатит». А платить за техническое задание на практике 99% таких вот хитрых компаний, абсолютно не готово. И уж поверьте, тот кто оплатит, тот на 100% эту работу закажет, так как понимает, что это обоюдовыгодный интерес.

Поэтому, или приступаем к работе при понимании грубых контуров и ожидаемого результата, при подписанном договоре, или делаем техническое задание, за которое получаем ПРЕДОПЛАТУ.

Никак по-другому. Ни коим образом не соглашайтесь, что-либо разрабатывать новым клиентам, с которыми не закрепились бизнес-отношения.

САМЫМ ЛУЧШИМ ПОДТВЕРЖДЕНИЕ ДРУЖЕСКОГО ОТНОШЕНИЯ И ДОВЕРИЯ СО СТОРОНЫ КЛИЕНТА, ЯВЛЯЕТСЯ ПРЕДОПЛАТА НА РАСЧЁТНОМ СЧЕТУ.

Сколько угодно можете бросать в меня помидоры и тапки, что мол результат не понятен, а деньги потрачены.

Господа, а вы, когда сотрудника на работу берёте, и он там что-то месяц ковыряется, но вы ж ему зарплату по законодательству заплатите?

Так вот и представитель ИТ компании бесплатно ничего делать не будет.

Если задаться вопросом, сколько заплатить за разработку технического задания?

Ориентировочно не меньше 10% стоимости будущей выполняемой работы в первой грубой прикидке бюджета.

Очень часто многие специалисты считают, что проработанное техническое задание – это как проект строящегося здания.

Соглашусь на 100%.

Но есть одно «Но».

Вы в курсе сколько стоит хороший проект? Поверьте – очень дорого.

И надо понимать, что:

во-первых, хорошее крепкое здание простоит 100 лет, а автоматизированный процесс прослужит в лучшем случае лет 15;

во-вторых, у строящегося здания, вырисованного на картинке, есть контуры, фасад, и как минимум понимание заказчика, как оно должно выглядеть по итогу;

в-третьих, при визуальном изображении, любой человек, ориентируясь на своё понимание вкуса и цвета, легче понимает где и что улучшить, или поправить.

Это далеко не все перечисленные преимущества. Самое главное нужно усвоить, что отношение к строительству недвижимого имущества несомненно гораздо серьезнее, хотя бы просто потому, что неправильно спроектированное строение может причинить ущерб для жизни и здоровья людей.

К ИТ проектам такого отношения нет, и там, где автоматизация несёт за собой высвобождение роли человека, как например автопилот, то такие проекты тянутся очень долго, с большим количеством испытаний и с очень огромными бюджетами.

Большие корпорации могут позволить себе потратить больше, просто хотя бы потому, что и желание получить от этого как минимум существенную экономию, и заработать больше, влечёт за собой приличные инвестиции.

Если проект не удался, ну что ж, спишут безнадёжную инвестицию с баланса компании и забудут, как плохой сон.

С небольшими компаниями дела посложнее, из-за скромного бюджета, прямого согласования собственника, ограниченности проекта.

Маленькая компания для серьёзной ИТ компании никакого интереса не представляет.

Помню как-то на одной из встреч сокурсников университета «BIG MONEY», один из представителей компании, заявившей о полторы тысячи программистов в штате, сообщил, что на проекты стоимостью меньше чем пятьдесят тысяч долларов они даже и не посмотрят.

А что может себе позволить собственник компании с ежемесячным оборотом в 30 000 долларов?

Конечно же мизер, который при стоимости разработчика начиная от 15 у.е. за 1 час, улетает в расходы и бьёт по карману этого собственника, так, что приходится искать на чём в этом месяце сэкономить, чтобы программисту заплатить.

Большие IT компании не хотят связываться с таким бизнесом не только потому, что бюджет скромный, а потому, что аппетиты получения за эти средства у заказчика ого-го какие большие.

У маленьких компаний способ выживания в большом мире бизнеса достаточно простой: «выжми по максимуму, отожди по полной». По-другому компания не выживет, она не может позволить себе разбрасываться налево направо своими оборотными активами, затопчут конкуренты.

У собственников амбиции не маленькие, рано или поздно хороший проект себя покажет и вырастет, но на этапе выпрыгивания из штанишек, которые становятся малы и хочется большего, иметь дела с такими предпринимателями дело весьма, как бы так поделикатнее сказать, «тонкое» и для нервов затратное.

Переговорный процесс в ходе обсуждения программных работ сразу упирается в вопрос: «А сколько это будет стоить?»

Цена, если она выше 100 условных единиц, всегда будет считаться априори «дорого» и подкрепится аргументом, что вот мол у таких-то «каких-то разработчиков» всё гораздо дешевле.

Ребята, но это же вам не китайские трусы на базаре выбирать. Вопрос качества явно не ощутим и его не разглядеть в микроскоп. И, очень важный момент, когда кто-то из программистов уже начинает работать с компанией, то его стоимость будет уже в разы ниже, из-за осведомлённости и ориентированности в базе данных.

Даже сторговавшись по сумме и приступив к работе, надо быть готовым, что без детально проработанного технического задания (так как его никто писать не собирался), IT компанию начнут выжимать как лимон, до последней капли сока.

Попытки отбрыкаться, увещевать, аргументировать, пусть даже явными и очевидными фактами, не помогут.

Собственник навалится на вас всем своим телом и будет требовать всё больше и больше, положи на стол свой козырный джокер: «НУ ВЫ ЖЕ ОБЕЩАЛИ, ЧТО ВСЁ БУДЕТ РАБОТАТЬ».

Что «всё» – не уточняется.

Всё – это значит ВСЁ, и точка, точнее две точки над буквой «ё».

Прокрутив киноленту назад до той самой точки и злополучной даты, когда вы договаривались о реализации технического задания, вспоминаем каверзный вопрос: «А оно будет всё работать», что вы ответили?

Ну конечно «да». Было бы странно, если бы вы ответили «нет».

Потому, что ваше понимание «всё» имело чёткие рамки и границы, которых у противоположной стороны явно не было, и контуры были размыты. А раз чётких очертаний границ дозволенного не нарисовано, то извольте батенька делать, пахать с утра до ночи.

Конечно, зайдя в тупик, и находясь в положении нависшей угрозы неоплаты того минимума на который компания согласилась (а теперь вы начинаете понимать, что законные требования на 50% предоплаты явно обоснованы), программисты ещё конечно попрыгают для достижения того самого эфемерного «всё и сразу», но эти прыжки к окончательному результату не приведут, так как включится ещё один фактор, название которого:

«сотрудники небольшой компании заказчика».

Если вы спросите меня есть ли разница между сотрудниками больших, средних, и маленьких компаний?

Есть, да и ещё какая.

Знаете – это как разница между большой организованной мафией с доном во главе, и шайкой хулиганов из неблагополучного района.

Культура компании, это главное отличие большой компании в работе, подходе, отношении и организации какого бы то ни было процесса, в ходе которого нужно стремиться к результату.

Порассуждаем о размерах самих IT компаний и как это влияет на выполняемые работы.

Большие IT компании – это своеобразное братство, где формируется своя культура, религия (кстати, большинство программистов явные атеисты), свой подход, своё отношение к жизни. Программисты в большинстве своём ещё те пофигисты, и в больших компаниях уровень пофигизма растёт пропорционально размеру этой самой компании. Раньше я считала, что программисты – это творческие люди, к сожалению, я ошибалась.

Почему к сожалению?

Потому, что количество ошибок, совершившихся по причине именно этого искажённого представления, прямо пропорционально потере денег, явившихся следствием именно этого заблуждения.

Творческие люди – это тело на 99,9%, заряженное эмоцией, в то время как разработчик вдоль и поперёк заряжен прагматизмом, на эмоции его не прошибёшь. Вот именно поэтому попытка доказать, например, что Бог есть, никак не укладывается в логику алгоритмических уравнений, которыми сплющ напичкана голова разработчика. Конечно, проще думать, что человек произошёл от обезьяны, и ведом по жизни базовыми инстинктами. Именно поэтому закон сохранения энергии так важен в IT сфере, в противном случае эмоциональное выгорание не за горами.

– Серьёзное отношение к чему бы то ни было в этом мире является роковой ошибкой.

– А жизнь – это серьёзно?

– О да, жизнь – это серьёзно! Но не очень...

«Алиса в стране чудес» Льюис Кэрролл

Маленькие IT компании – это небольшие демократические государства, тоталитаризмом тут не пахнет, каждый работник на вес золота. Главное, чтобы работа была, и тогда все довольны, что заняты.

Вполне вероятно, что разработчики делятся на три категории, это те, кто стремится работать сам на себя, те кто хотят работать в небольших компаниях, и те, кто хочет входить в большую огромную структуру.

Впрочем, вполне вероятно, в категории любых профессий можно разделить сотрудников поэтому же принципу.

Одно могу сказать точно, что в большой компании человек быстрее утомляется и выгорает. Работать в толпе, постоянно стремясь к неведомому идеалу, испытание не для слабонервных.

Постоянные подталкивания в спину, мол давай, беги быстрее, а то за всеми не успеешь, психологическое давление, постоянные соревнования и конкурсы, кто в компании круче.

Вы видели, как проходят беговые марафоны для любителей, когда вся толпа дружно рванула с места и потом разделилась на тех, кто быстрее, и на остальных догоняющих. Так вот, большая компания – это такой же марафон, только не на один день, и бегать в этом забеге нужно ежедневно без выходных и отдыха.

«Нужно бежать со всех ног, чтобы только оставаться на месте, а, чтобы куда-то попасть, надо бежать как минимум вдвое быстрее!»

«Алиса в стране чудес» Льюис Кэрролл

Хотя не исключаю момента, что люди разные и кому-то такие ежедневные нервные марафоны в кайф, но здоровье всё-таки у человека не железное и от такого бега, очень долго потом придётся лечиться.

Для программиста идеальным состоянием является состояние покоя и концентрации, чего очень сложно достичь, работая штатным сотрудником, но работая в одиночном режиме ситуация не лучше, надо искать клиентов, что может утомлять не меньше чем назойливые звонки товарищей по работе.

Для чего мы всё это тут философствуем, именно для того, что, когда любая из сторон обозначает своё намерение к сотрудничеству, необходимо обязательно сопоставить размеры, прежде чем подписывать договорные условия.

Сравним ситуацию выполнения одного и того же задания слоном и медведем. Где в роли слона будет большая корпорация, а в роли медведя одиночный программист.

Поставим одинаковую задачу, например, перетащить ветки поваленного крупного дерева в джунглях из одного места в другое.

Представили?

А теперь представьте, как каждое из животных будет выполнять свою работу.

Понимаете, что по-разному. А ещё им надо договориться о синергии и эффективной коммуникации.

Тяжело это будет, и в жизни практически не реализуемо.

Поэтому, принимая решения о старте проекта, очень важно сопоставить свои потребности и возможности с потребностями и возможностями другой стороны, дабы не получить потом полное разочарование из-за несостоявшегося результата и демотивированных людей.

Глава 3. Программисты не вечны, и вы тоже.

*«Если бы каждый человек занимался своим делом, Земля бы вертелась быстрее».
«Алиса в стране чудес» Льюис Кэрролл*

Не поймите мои слова превратно, однако в этой главе необходимо разъяснить вопрос, прежде чем вы кинетесь в омут реализации любого IT проекта.

Возомнить себя новым Стивом Джобсом легко, но вот потом удариться головой об суровую реальность твёрдую как бетон, очень больно.

*Знаешь, одна из самых серьезных потерь в битве – это потеря головы.
«Алиса в стране чудес» Льюис Кэрролл*

Однозначно, что тысячи компаний провалили тысячи проектов просто по причине того, что не провели предварительное исследование, Research.

Research – англ. исследование. Сленговое слово, означающее процесс поиска исходного материала, который в дальнейшем будет использован в проекте.

Research – подразумевает сбор информации по всем вопросам проекта, в т.ч. информацию о рынке конкретного продукта.

Задача такого исследования – анализ внутренней и внешней среды предприятия и подготовка на этой основе информации, необходимой для принятия решений, касающихся нового проекта

Без исследования не будет технического задания, конкретных сроков, и не будет определён хотя бы приблизительно бюджет.

Особенно остро этот вопрос касается планирования реальных сроков реализации, включая и оптимизацию кода с разработчиками, которые его, этот самый код написали.

Для начала поговорим о сроках, которые вы сами себе прикидываете перед тем как обозначить на календаре красным флажком дату окончания и получения результата проекта.

*План, что и говорить, был превосходный: простой и ясный, лучше не придумать. Недостаток у него был только один: было совершенно неизвестно, как привести его в исполнение.
Алиса в стране чудес Льюис Кэрролл*

В своей предыдущей книге: «Если бы программисты строили дом, или Как не потерять миллионы в IT-проектах» (если не читали, то конечно же рекомендую), я касалась вопроса нездорового оптимизма разработчиков в отношении сроков выполняемой работы. И это даже при том, что программист может быть с ого-го каким опытом, но это не мешает ему затянуть срок выполнения работы минимум в два раза, и это как «здрасьте».

Раньше я наивно полагала, что этой болезни подвержены только новички, но нет, даже те, кто в сфере программирования более 15 лет, всё так же выставляют крайний срок реализации, совершенно не соответствующий реальности.

Можно ругаться, психовать, угрожать, танцевать и разбрасывать деньги сверху, как арабский принц на танцовщицу, никакие способы вас не спасут от того, что сроки не будут выполнены.

ЗАПОМНИТЕ, ЧТО В 99,9999999% СРОКИ IT ПРОЕКТОВ НЕ ВЫПОЛНЯЮТСЯ, И ПЕРЕДВИГАЮТСЯ НА БОЛЕЕ ПОЗДНИЕ ДАТЫ!

Кроме нервного потрясения от вашего волнения вы ещё получите выгоревшего и немотивированного разработчика. Особенно опасным мероприятием является затея в последний день перед сдачей заказчику выжимать из программиста все соки до самого утра.

ВАЖНО ПОМНИТЬ, ЧТО В ТАКОЙ СИТУАЦИИ КОД НА СЛЕДУЮЩЕЕ УТРО НЕ ЗАРАБОТАЕТ.

И к тому же на следующий день чинить его будет некому, так как программист «лыка вязать не будет», будет находиться в полном бессознательном состоянии из-за усталости.

Советы СТОРОНЕ, предоставляющей услуги программистов.

Совет 1. Когда называете сроки выполнения клиенту смело прибавляйте ровно половину к сроку, указанному программистом. В противном случае вас ожидает нескончаемая череда дней, под названием «завтра всё будет готово», именно так вы будете отвечать каждый день клиенту, который будет вам звонить с вопросом, о том, когда будет закончена работа.

Завтра никогда не бывает сегодня! Разве можно проснуться поутру и сказать: «Ну вот, сейчас наконец завтра»?

«Алиса в стране чудес» Льюис Кэрролл

Совет 2. При приближении вашей оптимистичной даты, минимум дня за три, адекватно оцените ситуацию и ещё раз обновите дату. Обязательно предупредите клиента, как бы это не было тяжело. Знаю, знаю, не приятно. Но за это вас не расстреляют, а на будущее вы спасёте себя от полного провала в глазах клиента в последний вечер, когда надо сдавать работу, а сдавать по факту нечего.

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.