

Иван Белокаменцев

Это всё, программисты?

СОДЕРЖИТ
НЕЦЕНЗУРНУЮ
БРАНЬ

18+

Иван Белокаменцев

Это всё, программисты?

<https://litres.ru/74445398>

SelfPub; 2026

Аннотация

Профессия программиста стремительно исчезает.

Этот процесс уже не остановить. Начался он давно - с бурного расширения отрасли ИТ, породившей огромную волну миграции "войти в АйТи". Продолжился процесс разделением труда - профессию программиста растащили на мелкие кусочки. Большие континенты на карте профессий ИТ распались на архипелаги из мелких островков.

Сейчас на ИТ налетело цунами ИИ. Мелкие островки ИТ-профессий уходят под воду и исчезают. Держатся лишь те, что покрупнее. Те, кто более программист.

Кто же он такой, этот программист? Ответ есть в книге.

Книга - сборник рассказов о программистах и профессии, написанных за 10 лет. Во времена, когда отрасль ИТ росла, а профессия программиста - исчезала.

Актуальны ли рассказы сейчас? Помогут ли они спастись от цунами ИИ?

Автор не знает. Пусть решает читатель.

Но вспомнить о том, как оно было - точно помогут.

Содержание

Программист на больничном	4
Выйди и зайди правильно	20
Программировали два товарища	30
Леньхаки. Так лень, что не лень	37
След в туалете	45
Клуб использованных техлидов	53
Реверс-хантинг	63
Пусть программирует Серёжа	69
Конь остановлен, изба догорела	78
Начинающие программисты боятся темноты	83
Заградотряд	93
Иррационально свободный	103
Крепостное право в ИТ	112
PROграммист со звёздочкой*	121
Что вижу, то и программирую	124
Конец ознакомительного фрагмента.	126

Иван Белокаменцев

Это всё, программисты?

Программист на больничном

— Куды? – грозно спросила бабка, встав спиной к двери.

— Куды надо. – передразнил неопрятный парень, с красными глазами, взъерошенными волосами и в черной футболке с надписью «Король и Шут».

— Ну-ка, не пушай его! – встала со скамейки другая бабка. – Ишь, вумные, без очереди лезут! Мы тут с шости утра стоим!

— Не стойте, а сидите. – ответил парень. – Дайте пройти, я по делу.

— Знаем мы, по какому ты делу! – вступила симпатичная девушка лет тридцати. – Только спросить?

— Да что с вами такое, а? – возмутился парень. – Работаю я здесь! Я – программист!

— Давай, рассказывай. – улыбнулась девушка. – А халат где?

— На портянки пустил. – в тон ответил парень. – Дайте пройти.

— Не пущу! – бабка у двери, похоже, не все слова поняла. – Хоть программист, хоть анархист, не пущу!

— Слушай, бабка... — заговорщицким тоном сказал парень, наклонившись поближе к слуховому аппарату. — Я щас до интернета дойду, и талон твой аннулирую. Пусти, говорю!

Дверь за бабкой попыталась открыться, но уткнулась в костлявую спину.

— Мадам, в сторону, там кто-то выйти пытается. — улыбнулся парень.

— Ой, а я... — бабка поспешно посторонилась.

Из двери выглянул врач — молодой парень, в очках, лысый, с татуировкой на шее. Рукава халата были закатаны, и предплечья тоже были в татухах.

— Свят, свят... — попятилась назад бабка.

— Что вы тут устроили? — закричал врач. — Я вот сейчас как устрою кварцевание на полчаса, попляшете у меня! Руслан, заходи!

— А вы можете побыстрее принимать? — крикнула вслед закрывающейся двери девушка, но ее никто не услышал. — По полчаса принимают, кошмар...

В кабинете никого, кроме врача и программиста, не было.

— Что тут у тебя, Вась? — спросил Руслан, усаживаясь за стол, на котором стоял компьютер.

— Да зависла, тварь, система эта долбанная. — Василий был очень раздражен. — Напридумывали говна какого-то, бездари, зла не хватает.

— Ну, систему нам из министерства назначили. — пожал плечами программист. — Ща починим, не ори.

Через несколько минут Руслан закончил. Помог старый дедовский метод – профилактическая перезагрузка.

— Готово. А чего не принимаешь? – спросил Руслан, вставая со стула.

— А смысл? – устало сказал врач. – Сейчас правила такие – надо сразу, на приеме, вбивать данные, иначе вздуют потом. Не больница, а дерьмо собачье. Скоро все врачи разбегутся, надоело себя идиотом чувствовать. Не с пациентом общаешься, а данные вколачиваешь, как крыса офисная какая-нибудь.

— Давай я повбиваю. – пожал плечами Руслан.

— В смысле? – удивился врач. – Серьезно?

— Ну там сложное что-то?

— Да какое там... — махнул Вася рукой. – Одно и то же, каждый раз, по каждому пациенту. Как родился, где учился, стул твердый или нет, достало уже...

— Ну давай, мне один хрен заняться нечем.

— Следующий! – громко крикнул врач.

В дверь быстро вбежала давешняя бабка. Увидев Руслана, остановилась, как вкопанная.

— А ентот, антихрист, чего тут делает? – показала она костлявым пальцем на программиста.

— Это, бабуля, представитель пенсионного фонда. – успокоительно сказал врач. – Пришел с инспекцией, проверяет, как мы, сатрапы изуверские, с пожилыми людьми обращаемся.

— Ага! – обрадовалась бабка. – Все природы в одном месте собрались! Ну я вам чичас...

— Раздевайтесь! – строго сказал врач.

— Твою мать... — сконфузился Руслан, и, как мог, поближе прильнул к монитору, чтобы не видеть разворачивающегося – действия и серой вязаной бабкиной кофты.

Через час, когда вышел очередной пациент, вдруг стало очень тихо. В дверь никто не заходил, и даже не заглядывал. Врач выглянул в коридор, повертел головой в разные стороны, цокнул языком и вернулся в кабинет.

— Рассосались. – пожал он плечами. – До конца смены еще два часа, чем заняться-то?

— Ну и работка у вас. – встал со стула Руслан. – Какой только фигни не наслушаешься.

— А мне понравилось. – улыбнулся врач. – Представляешь? Первый раз, наверное, понравилось. Почувствовал себя врачом, а не машинисткой. Угроздило же в терапевты пойти... Слушай, может ты и завтра придешь?

— Данные вбивать? Нет уж, увольте. У вас тут системные проблемы, а не врачебные.

— Ладно, не ной. Куда двинешь?

— К себе.

Руслан вышел в коридор, и обомлел от дисбаланса. В кабинет Василия не было ни одного пациента, зато в остальные, мать честная... Немного подумав, вернулся обратно.

— Давай остальных тоже примем? – предложил он. – Так

же можно?

— Можно. – пожал плечами врач. – Только нафига?

— В рамках эксперимента.

— Ну иди, обрадуй толпу. – улыбнулся Вася.

Руслан улыбнулся в ответ, высунулся в дверь и громко крикнул: «Свободная касса!». Ожидаемой реакции не последовало, только люди стали пугливо озираться на небритую, с всклокоченными волосами, орущую морду.

— Граждане, подходим, не стесняемся! – немного подумав, снова закричал программист. – Свободный терапевт! У кого нулевые талоны, все сюда! Если, конечно, вам не принципиально, кто ваши несчастные души будет лечить.

— У меня нулевой! – от толпы отделилась женщина лет сорока. – С утра стою, не пропускают, только по записи!

— Велкам! – Руслан услужливо приоткрыл дверь, пропуская даму.

В толпе началось брожение, и вслед за женщиной к кабинету подошли еще несколько человек – очередь сформировалась.

До конца смены успели принять еще порядка двадцати пациентов. Руслан быстро вбивал ничего не значащую информацию, а потом, под диктовку врача, записывал диагноз и рекомендации по лечению.

Когда смена закончилась, врач – усталый, но довольный – плюхнулся на кушетку.

— А гримза тебя не съест? – спросил Руслан.

— Не. — махнул рукой врач. — У нее сейчас такой геморрой, что обедать врачами нет ни сил, ни желания.

— Ты про что?

— Ну эта, хрень, как ее... Бережливая больница.

— А, да, слышал.

— Ну вот. Срок внедрения почти вышел, а ни хрена не сделано. На днях проверка, она сегодня с утра орала на нас, ну, что мы... Как это сказать-то...

— Рукожопые?

— Да-да, вроде того. Я слышал, что ее снимут, если проект неудачно закончится.

— Странно это...

— Что именно?

— Бережливая больница. Это, я так понимаю, Lean?

— Чего?

— Ну бережливое производство, Lean. Наверное, кто-то шибко умный решил, что те же методы можно применить в больнице. На производстве не получается, в логистике не получается, так в больницах приткнуть решили. Все никак не сдохнут.

— Кто? Министерские?

— Нет, консультанты эти долбанные. Которые ничего, кроме Лина, не знают. Нигде не могут добиться никакого полезного результата, вот и ищут себе новые рынки сбыта. До госструктур вот добрались.

— А ты чего, сечешь в этом? — прищурился врач.

— Пф, чего там сечь-то... Я ж до больницы на заводах работал, и в логистике. Навидался этого всего. Скоро, того и гляди, в больницах будут скрам внедрять. Или сбалансированную систему показателей, или 5S.

— Это чего?

— Да хрень на палочке. Заставят тебя рабочее место в порядке держать. Слева – палочки, которые в горло толкать, справа – карта пациента, сверху – фонендоскоп, и так далее.

— Кто ж фонендоскоп на столе держит. – улыбнулся врач.
– Все приличные люди его на шее носят.

— Ну, когда придут 5S внедрять, скажут на стол положить. Или в кармане таскать – обязательно в левом. Такая методика.

— Дичь какая-то...

— Э, это ты на заводах не бывал... Там, бывало, с работы выгоняли мужиков, которые метчик в кармане таскали. Потому что положено его на станине держать. А он оттуда, сука, падает все время, от вибрации, и прям в эмульсию. А консультантам этим что в лоб, что по лбу – говорят, так положено, держите на станине.

— Мда... Ну вот у нас же с тобой красиво получилось сегодня? Давай завтра повторим, а? Блин, Русля, реально... С меня пузырь вискаря, если завтра придешь.

— Да приду, че мне. – пожал плечами программист. – У нас с тобой нормальный ТОС получился.

— Чего?

— Щас расскажу.

Утром Руслан спокойно сидел за своим столом и пил чай. Вася работал во вторую смену, и до обеда заняться было особенно нечем.

Внезапно зазвонил телефон. Руслан взял трубку, и услышал Васю – тот велел срочно бежать в кабинет к главврачу.

— Да говорю вам, это реальная тема! – услышал Руслан, войдя в кабинет – это Вася очень активно в чем-то убеждал Наталью Петровну.

— Вася, ну слушай. – устало отвечала главврач. – Мне не до этого сейчас, сам знаешь, скоро проверка, меня повесят, если показатели принятых на специалиста не вырастут. Кто только эти нормы придумывает... Пятнадцать минут на одного для терапевта, да с внесением данных в компьютер...

— Так и я о том же! Я вам о чем рассказывал-то битых полчаса?

— Про ТОС какой-то...

— Нет! Знаете, сколько я вчера пациентов принял за смену?

— Чего знать-то, я сейчас посмотрю... — Наталья Петровна стала копаться в компьютере. Вася замолчал, удовлетворенно улыбаясь. Руслан, пользуясь паузой, тихо поздоровался и присел на свободный стул.

— Тридцать три?! – округлила глаза главврач. – За одну смену? Это сколько получается...

— Чуть больше семи минут на пациента. – подсказал Рус-

лан. — Но, если быть точнее, вдвоем мы работали не с утра, а последние часа три, и приняли... Сколько?

— Тридцать. — подхватил Вася, доставая смартфон и судорожно ища калькулятор. — Так, это сколько получается...

— Сто восемьдесят на тридцать поделить не можешь? — улыбнулся программист. — Шесть минут. А с утра ты принял трех пациентов, за час. Двадцать минут на душу населения.

— А ты их принимал вообще? — нахмурилась главврач. — Или так, поделил аспирин на две половинки, эта для головы, эта для жопы, и не перепутайте?

— Да я никогда так не принимал! Это ж кошмар, когда сидишь, как орел двуглавый — и с пациентом надо разговаривать, и данные эти вбивать, да еще про норму все время думать! А тут я только делом занимался, ничего лишнего. Знаете, какие они довольные уходили? Особенно те, кому на работу надо.

— Какую работу? Они ж больные.

— Больные-то больные, но у нас ведь капитализм. Больничные берут только по уходу за ребенком, остальные сразу на работу бегут. А нулевые так вообще меня чуть не расцеловали.

— Ладно, я поняла... — задумалась Наталья Петровна. — А в чем смысл? Почему так получилось?

— Руся, давай!

— Чего давать?

— Объясняй!

— А... — сконфузился программист. — Это ТОС, теория ограничений систем. Прием врача — это бизнес-процесс такой, и к нему...

— Чего? — нахмурилась главврач.

— А, да... — осекся программист. — Ну, процесс, короче. Пришел пациент, надо выполнить определенные действия, так? Спросить, чего у него болит, послушать, горло посмотреть там, ну вы лучше меня знаете.

Главврач кивнула.

— Ну вот. А еще надо данные в систему вбить. Потом распечатать бумажку, вклеить в карту, пациенту написать рецепт и сказать, когда прийти в следующий раз. Такой процесс.

— Ну да, это я знаю. А где здесь твой ТОС?

— У любого процесса есть ограничение, узкое место, бутылочное горлышко. То, что делается медленнее всего.

— А, поняла. — кивнула главврач. — В нашем случае это — вбивание данных в компьютер?

— Да. И скорость всего процесса — приема пациента — ограничивается скоростью вбивания данных. Именно это действие занимает больше всего времени.

— И что дальше? — в голосе главврача появился интерес.

— Дальше просто. Узкое место мы определили, и у нас два варианта — либо максимально его защитить, либо пересмотреть процесс так, чтобы часть нагрузки ушла с ограничения.

— И что из этого сделали вы? А, поняла... Ты сел вбивать данные, тем самым разгрузив Васю?

— Ну да. Убрали непрофильную нагрузку со специалиста. Теперь у процесса новое ограничение – Вася.

— Я? – вмешался врач. – Со мной-то что не так?

— С тобой все в порядке, ты – ключевое звено процесса. Смотри сам. Как бы мы тебя не защищали, не разгружали, ты все равно должен с пациентом поговорить, послушать его там, горло посмотреть и так далее.

— Ну да.

— Вот. Эти действия выкинуть нельзя, поэтому скорость приема теперь будет определяться скоростью выполнения этих действий – обычных, нормальных, врачебных процедур. Фишка в том, что, кроме скорости приема, теперь ничто не мешает – нет лишних звеньев, вроде вбивания данных. Хотя...

— Что? – подхватила главврач.

— Теория ограничений, а точнее – ее создатель – говорит, что всегда можно сделать быстрее и лучше. Вот, например...

Руслан задумался на несколько минут. Остальные молчали, понимая, что могут сбить с мысли.

— Да, вспомнил. – улыбнулся программист. – Вот раньше, насколько я помню, у врача была медсестра, когда еще не было у вас компьютеров. Она делала примерно то же самое, что и я вчера – снимала непрофильную нагрузку с ограничения, то есть с врача. С карточкой работала, может даже

горло смотрела...

— Нет, горло всегда врач смотрит. — покачала головой главврач.

— Ну ладно, не суть. Наверное, что-то еще можно придумать...

— А зачем? — не выдержал Вася. — Мы итак с тобой, без подготовки и всяких там систем и ограничений, по шесть минут принимали!

— Руслан, а как-то можно это попробовать на других врачах?

— Руслан мой! — вскричал Вася. — Он мне обещал, что сегодня будет со мной всю смену!

— Нет, так не годится. — скривилась Наталья Петровна. — Руслан — важный специалист, мы не можем использовать его для такой работы. В конце концов, мы платим ему серьезные деньги.

— Не смешите мои тапки. — улыбнулся Руслан. — С каких пор пятнаха стала серьезными деньгами? Вы Васе минимум вдвое больше платите.

— Так, давайте не будем спорить о том, что нам неподвластно. — строго сказала главврач. — Ставки утверждает министерство, а не я.

— Да я не против, знал, на что шел. — пожал плечами программист. — Но я не против данные повбивать, если для пользы дела.

— Ладно. Но ты же один.

— Нет, не один. Вы по коридорам пройдите, в кабинеты загляните. Особенно те, что вокруг вашего, где всякие административные работники сидят. Куча народу, которым весь день заняться нечем. И все неплохо с компьютером обращаются.

— Например?

— Леночка.

— Хм... Лена – мой секретарь, у нее много работы.

— Это она вам так сказала? – улыбнулся программист. – Я, если честно, задолбался уже с ее компом возиться. Каждый раз, когда интернет пропадает, истерику поднимает.

— Почему? Документы отправить не может?

— «Сокровища пиратов» в оранжевой соц.сети не работают. – улыбнулся Руслан. – Короче, поверьте мне на слово, времени у нее – вагон и маленькая тележка.

— Лена! – крикнула главврач. – Зайди!

В дверь вошла молодая, очень симпатичная, крашеная рыжеволосая девушка в белой блузке, короткой юбке и на каблуках.

— Да, Наталья Петровна, я здесь. – томно сказала она. – Чаю?

— Ты с компьютером дружишь?

— Да, конечно. – кивнула Леночка, подозрительно глядя на Руслана. – А что?

— Сегодня пойдешь... Так, к кому... К Бабаеву, что ли?

— Пусть с нами пойдет. – сказал Руслан. – Под моим над-

зором у Васи поработает. Проведем эксперимент.

— Куда? Зачем? – заволновалась Леночка.

— Отличная идея, Руслан! – поддержала главврач. – Так, Лена. Иди с парнями, будешь делать то, что скажут. Вбивать данные по приемам. От меня сегодня никаких других задач не будет.

— Что? – вскричала Леночка. – Почему я? Я не медсестра, а офис-менеджер!

— Слушай! – с напором сказала главврач. – Ты же в курсе ситуации? Скоро проверка, и если все пройдет плохо, меня с должности снимут. Знаешь, кого поставят на мое место?

— Нет... — в голосе Леночки прозвучала неуверенность.

— Этого, как его... Эффективного менеджера, как в сто восьмой. Вот и будете, на пару – эффективный менеджер и офис-менеджер. Смотрела «Аритмию»? Так тебя менеджержить будет, что юбка по швам затрещит. Вспомнишь еще меня добрым словом.

— Да я, Наталья Петровна... Вы – самый лучший руководитель, с которым мне работать приходилось. – затараторила Леночка. – Я для вас...

— Ты для меня пойдешь с Васей и Русланом, и будешь делать то, что они скажут. Это только на сегодня. Поняла?

— Да, поняла! – кивнула Леночка. – Ребята, куда идти?

— Вась, давай, проводи. Руслан, ты задержись.

Вася неопределенно улыбнулся, встал со стула и пошел к двери. Леночка засемила следом.

— А дальше что? – спросила главврач, когда дверь закрылась.

— Посмотрим, как сегодня пойдет. Леночка нужна только для эксперимента, будем пробовать других. Со стола справок снимем, их там три, девицы-то. Все равно сидят ничего не делают, туда больше ругаться приходят, чем спрашивать.

— О чем ругаться?

— О долгом приеме, о хамстве, «где тут у вас начальство, позовите немедленно». Еще много жалуются на электронную очередь – ну, в которой надо брать талон, чтобы взять талон.

— Это вообще засада. – сокрушенно сказала главврач. – Заставили ее установить...

— Да знаю я.

— Но я не об этом. Что дальше-то? На следующей неделе министерские припрутся, смотреть, что я им скажу?

— А чего они смотреть будут?

— Показатели, в первую очередь. По врачам пройдутся. Думаю, будут докапываться про бережливую больницу. Там такой талмуд, я даже дочитать не смогла. Этот твой ТОС – это же не бережливая больница?

— Ну, и то, и другое – абстракции.

— Чего?

— Блин... Универсалии. На латыни, наверное, *universalia*?

— *Universalis*. – поправила главврач. – Ты к чему это?

— Упакуем одно в другое, они вряд ли разбираются. ТОС

и бережливое производство похожи, если смотреть только на внешнюю сторону. Кардинальные отличия в философии, но кто эту философию в министерстве понимает.

— Ты сможешь это, как ты там сказал...

— Упаковать.

— Да.

— Смогу. — кивнул Руслан. — Делов-то. Главное, ведь, показатели?

— Да.

— Ладно, сделаем. Вы девушкам с ресепшена скажете, чтобы слушались?

— Откуда?

— Блин, терминология... Из стола справок.

— Да, прям сейчас. Пойдем вместе.

Выйди и зайди правильно

Управлять программистами в ИТ-компании – дело особое. Не сказать, что сильно сложное. Совру, если назову простым. Особое.

Книжек, статей, курсов создано великое множество. Что-то помогает, у кого-то получилось. Собственно, промахнуться с оценкой сложно – отрасль ведь не просто существует, но и весьма активно развивается. Даже в нашей деревне. Значит, как-то, кое-как, с горем пополам, но управление программистами не только возможно, но и получается.

Однако, кризисы случаются. Бывает, что затяжные – например, какая-нибудь серьёзная деревенская компания не может ни найти, ни воспитать приличного руководителя программистов. Благо, сейчас это не великая проблема – бизнес вполне себе растёт лишь потому, что растёт рынок, и заполнить его некому. Поэтому за начальника сойдёт и табуретка. Но – лишь до поры, до времени.

Регулярно ИТ-бизнес думает-думает, решает-решает, плюнет, да и снова попробует старый добрый метод: нанять успешного руководителя из другой сферы. Производство, строительство, торговля – не суть. Оттуда много кандидатов, у них подтверждённый опыт, зачастую – MBA, и они такие прям чёткие всегда! Знают, чего хотят, на что способны, чего им нужно.

Мне этих руководителей всегда искренне жаль – я знаю, что их ждёт, каков будет их путь, почему они в итоге уволятся. Я не семи пядей во лбу, просто в течение многих лет наблюдал их в обеих сферах – ИТ и той, из которой они пришли, где добились успеха. Могу плюс/минус сопоставить контексты, работающий инструментарий, важные и второстепенные личностные качества.

Много раз пытался объяснить, помочь – и начинающим руководителям-варягам, и тем, кто их тащит в ИТ. Никто, разумеется, не слушал. Ну да я уже привык, поэтому сижу себе помалкиваю.

Однако, тут такая история вышла: в ИТ-компанию собралась устроиться некто Ира, хорошая знакомая моего хорошего знакомого. Он, как мог, пытался её отговорить, но не преуспел, т.к. сам работает программистом, и руководителем никогда не был.

Попросил меня. Я предложил формат статьи – и моё мнение услышит, и ваше. В конце концов, вполне возможно, я не прав, как это обычно случается (если верить вашим комментариям).

Итак, погнали. Буду писать, как бы лично обращаясь к Ирине.

Кто ты?

Привет, Ира. Я примерно знаю, кто ты в плане работы – наш общий знакомый рассказал. Я очень хорошо знаю три из четырёх предприятий, где ты работала руководителем – го-

род у нас маленький. Стиль управления, который там в ходу и делает человеку успех, мне хорошо знаком.

Чтобы ребятам, читающим текст, было понятно, я скажу: ты работала руководителем службы снабжения производственного предприятия, чуть раньше начальствовала над экономистами в компании, специализирующейся на кооперации, ещё рулила сервисной службой производителя промышленного оборудования, ну и был опыт построения службы менеджмента качества – с неё началась твоя карьера.

Тебе везде сопутствовал успех. Нет: ты создавала успех сама, максимально чётко соответствуя требованиям и настроениям контекста. Однако, в ИТ тебя действительно ждут проблемы. В части управления тут всё немного иначе.

Подчинение

Там, где ты работала – руководителем среднего звена на, скажем так, «обычных» предприятиях – с подчинением проблем, по большей части, не было. Разумеется, встречаются иногда перцы, которые увлекаются саботажем или «итальянкой», но именно иногда.

В основном «подчинение», как актив, принцип, фора - у тебя было. Управляя людьми, придумывая изменения, определяя правила, очерчивая границы, ты опиралась на подчинение. Ты, пожалуйста, сейчас не бросай читать, а вдумайся в эти слова. Люди тебе подчинялись. Ты настолько к этому привыкла, что давно перестала замечать.

Наверное, проблема подчинения у тебя возникала в пер-

вые дни или недели работы в новой должности или компании. Но ты знала, как её решить. Например, в мягкой или жёсткой форме обозначив: кто не подчиняется – работать здесь не будет. Мы все прекрасно понимаем, что линейный персонал в отдел закупок, экономики или службы менеджмента качества найти – не великая проблема, поэтому риск оправдан.

Равно как мы знаем, что в подавляющем большинстве случаев никто не бежит из отдела при появлении нового руководителя – сначала присмотрятся, приценятся. Вдруг просто границы обозначает, а то и сам сольётся. Это игра такая, кто страшнее зарычит.

Так вот, в ИТ про подчинение забудь. По крайней мере, управляя программистами. Никто не будет тебе подчиняться лишь потому, что ты – начальник. Тут всё гибко, зыбко и неопределённо. Ситуация, когда толковый программист ценится компанией выше, чем его начальник – рядовая, а не из ряда вон. Причём, я говорю и о денежной, и о стратегической ценности.

Иногда будет казаться, что тебя наняли гувернанткой к капризному барчуку. Это не обманчивое впечатление. Да, иногда надо проявлять строгость – но очень и очень ограниченную по твёрдости и возможным негативным последствиям. И ты, и программист будут понимать: твоя строгость – показная, не настоящая. Игра такая.

Я поставил подчинение на первое место, потому что, по

моим личным наблюдениям, это главное отличие управления в ИТ и «во всём мире». Ты наверняка читала книги по менеджменту, особенно классическому, и не могла не заметить, что тема подчинения там... Практически не затрагивается. Более или менее ей уделяют внимание в книгах по лидерству, но в несколько ином ключе – как вдохновить и повести за собой. Это уже тема мотивации, ниже про неё напишу.

Так вот, приходя в ИТ, готовься управлять, не приказывая. Учись – так быстро, как сможешь – мягким методам управления. Жёсткими ты добьёшься только одного – быстрого выхода на рынок труда.

Авторитет

Наверное, как-то уныло и безрадостно прозвучало, про подчинение-то... На самом деле, получить в свои руки высокую управляемость можно, но этот путь тебе вряд ли подойдёт – надо стать авторитетом среди программистов. Полагаю, таких планов у тебя нет, ибо понадобятся годы обучения.

Принципиально есть два варианта обретения авторитета. Первый, основной, наиболее надёжный, как написано выше – стать профессионалом в программировании. Стезя у нас сложная, интеллектуальная, постоянно меняющаяся в плане технологий и подходов, поэтому уважение, в первую очередь, базируется именно на профессиональных качествах.

Знаю, на «обычных» предприятиях можно стать авторитетом другими способами. Например, знакомством с дирек-

тором или его секретаршей, умением продвигаться по карьерной лестнице с опережением естественного хода событий, способностью договариваться со смежными службами и т.п. У нас, в ИТ, всё это тоже имеет значение, но не для программистов, а для твоих потенциальных коллег – руководителей.

Второй принципиальный путь завоевания авторитета, который пока ещё востребован в провинции – «руководитель-отец» или «руководитель-мать». Умение создавать комфортную, защищённую среду для программистов. В принципе, прочитав пару книг, ты сможешь переупаковать и продать этот навык, как «опыт фасилитации», «работу скрам-мастером, защищающим команду».

Почему в провинции – тут до сих пор есть от кого защищать программистов. В том числе – от вышестоящего начальства, которое иногда ещё мыслит старыми стереотипами, считая себя чуть ли не меценатом, благодетелем. Ну и пользуются тем, что многие программисты не могут, не хотят, или пока просто не решаются уехать или перейти на удалёнку.

Найдёшь таких, будешь их защищать – можешь стать авторитетом. Правда, получишь ложку дерьма в карму – пока ты их защищаешь, они будут под этими «меценатами» работать, просто под зонтиком.

Мотивация

Запомни, пожалуйста, одно слово: «интересно». Это –

главный мотиватор программиста, начиная с определённого момента развития, когда достигнуты два важных порога – «начало получаться» и «денег хватает».

Программисту должно быть интересно – язык программирования, фреймворк, проект, клиент, команда, офис, удалёнка. Ты, в конце концов.

Разумеется, остальное тоже важно. Ну, или, как минимум, имеет значение. Но – именно «тоже». Многие совершают ошибку – начинают с этих «тоже», когда не получается сделать «интересно». Да что уж греха таить – когда даже понять это самое «интересно» не выходит. А потом удивляются, чего программисты бегут без видимой причины.

Раньше ты имела дело совсем с другой мотивацией. Ключевое, пожалуй, отличие: программистам нравится дело, которым они занимаются. Не работа – компания, должность, проекты – хотя это тоже важно (ох уж это «тоже»). Именно дело – программирование, разработка, проектирование, отладка, ковыряние, запуск и прочие слова (их сейчас много придумали).

Предположу, что не ошибусь: на прошлых работах твоим подчинённым была интересна, скорее, работа, а не дело. Ничего не поделаешь – мало на свете увлечённых менеджеров по закупкам, экономистов, оформителей процессов по ИСО 9001. Поэтому основным инструментом, или объектом воздействия для создания и поддержания должной мотивации для тебя была работа, как место, а не как дело.

Для твоих «бывших» всё, что их окружает в офисе – люди, стены, столовая, корпоративы, кофемашина – были своего рода анестезией от дела. Ну, чтобы легче заходило и делалось, безболезненно, как по маслу.

С программистами иначе: всё это не должно мешать делу. Если ещё и поможет – прекрасно.

Не пытайся заменить «интересно» деньгами. Очень, очень многие твои предшественники на этом погорели, пытаясь предстать перед вышестоящими в выгодном свете. Например, ввязываясь в работу с Крупными Холдингами, где Много Денег. Там не интересно.

Демотивация

Знаю, ещё пользуетесь – там, на «настоящих» предприятиях. Штрафы за опоздания, просрочку задач, лишение премии службы качества за невыполнение плана продаж коммерческим отделом, порицания за дресс-код, невежливость, «не так посмотрела» и т.д.

Так вот – забудь. Нельзя. Табу.

В первые три месяца, а то и полгода – вообще никакой демотивации, пока не освоишься, не станешь хоть немного своей. Потом можешь мягко, нежно, заниматься самой лайтовой демотивацией – лишением мотивации. Да, ты уже поняла – пересаживанием человека на время с «интересного» на «неинтересное».

Планирование

Там, где ты работала, можно было довольно точно прогно-

зировать и планировать работу людей. Причина очень простая – была достаточно точно известна скорость выполнения тех или иных действий, операций, поручений. Не настолько, как у станков, но с вполне адекватной погрешностью.

Дело в повторяемости задач – так или иначе, но твои сотрудники делали примерно одно и то же. Оформляли документы – с известной скоростью печати, заказывали материалы у поставщиков – с понятной интенсивностью работы в КИС и документооборота с контрагентом, готовили отчёты – одни и те же каждый месяц.

У программистов всё иначе. Повторяемости мало. Если повезёт, можешь наткнуться на какой-то ИТ-конвейер – таких сейчас, в общем-то, достаточно много развелось. Что-нибудь вроде автоматизации мелкой розницы, установки готовой программы для бухгалтерии или запуска типовых сайтов. В принципе, всё это тоже считается ИТ, но там нет или почти нет программистов – именно на конвейере. Рядом, в соседней комнате, программисты вполне могут обнаружиться, но там повторяемости не увидишь.

Стоит ли?

Я не пытался тебя отговорить, или создать видимость чрезвычайной сложности управления в ИТ. Наверное, суть и цель моего послания ровно такая же, как в аналогичных рассказах для тех, кто хочет «войти в ИТ», но как программист.

Тут интересно, прикольно, классно. Но: иначе. Это надо понимать, принимать, и не формировать в душе ожиданий.

В твоём случае это будет сложнее, чем для программистов, потому что ты искренне считаешь, что просто меняешь объект управления, а субъект – тебя, Иру – существенно корректировать не надо. Ты идёшь, увешанная с ног до головы медалями прошлых достижений.

P.S.

Но результат того стоит. Если будешь воспринимать своё турне по управлению в ИТ не как необъяснимые трудности, провал или «сранные тупые ленивые хипстеры», а как полезный, другой опыт.

Другую, незнакомую, непонятную, но очень увлекательную культуру управления в мире, которым увлечён весь мир.

Программировали два товарища

Эта грустная история случилась на моих глазах. Не берусь утверждать, что она уже совсем прям завершилась – мало ли, вдруг кто-то из её героев прочтёт, задумается, что-то изменит... Ну или плюнет мне в кружку.

Итак, пришли в компанию, почти одновременно, два прекрасных программиста. Одного поля ягоды – опыт, квалификация, профессиональная любознательность, энергия. Раньше их карьеры развивались по одинаковому, иногда – равному треку (работали вместе).

Компания в нетерпении потирала ладошки, предвкушая результаты, развитие и непоколебимую дружбу двух программистов. Да, давайте их как-нибудь назовём – Коля и Вася. Всё было хорошо. Беда на горизонте даже не отсвечивала.

Весна

Это было золотое время. Программисты были просто прекрасны. Быстро влились в коллектив – благо, был опыт работы и, соответственно, навыки ассимиляции в совершенно разных средах. Проекты, клиенты, задачи – всё было и знакомо, и привычно. Даже немножко скучновато – в компании до того момента не было программистов столь высокой квалификации, по сумме параметров.

Менеджеры, руководители проектов, заказчики – все тащились. Коля и Вася не знали поражений, брали и решали

любые задачи. То, что раньше вызывало опасения, проблемы, просрочку и конфликты, теперь просто программировалось, сдавалось и оплачивалось.

Если кратко описать, чем именно занимались наши герои, получится так: самым сложным программированием.

Но потом случилось непоправимое – опыт Коли и Васи решили использовать шире. Их, как положено, попросили заняться чем-то между наставничеством и управлением.

Лето

Итак, обоим дали в одну руку джунов, в другую – карт-бланш. Цель, если можно так выразиться – парни, сделайте что-нибудь классное. Вы такие, прям такие, что слов нет. Хотим, чтобы вы не просто и не только программировали, но и как-то ещё наследили.

Ни у Коли, ни у Васи особого опыта работы с подчинёнными не было. Советовать им, разумеется, никто не брался – слишком уважаемые были личности. Пришлось нашим героям самостоятельно решать, куда и как двигаться.

Коля, поразмыслив, решил ничего особо не менять в структуре работы и направлении движения. Как занимался сложным программированием, так и продолжил, но уже не в одиночку, а с командой джунов. На том и решил построить успех своего новоиспечённого отдела.

Как это примерно выглядело. Коля сам общался с заказчиком – а это мог быть как клиент, так и руководитель проекта – после чего, как мог, объяснял джуну задачу, погружал

в контекст, историю работы с клиентом и во всё, что считал нужным. Молодые программисты, получив такие задачи, искренне и честно пытались их решить.

Что-то получалось, что-то нет, но Коля не унывал. Объяснял ещё раз, помогал, общался с заказчиком на всём протяжении жизненного цикла задачи, включая конфликтные моменты.

Результаты первых трёх месяцев у Коли были просто великолепные. В цифрах, если грубо, команда выдала математически красивый результат: каждый джун производил столько же, сколько Коля в одиночку.

У Васи было хуже, сильно хуже. Он почти сразу выбросил из командного продуктового портфеля решение сложных задач, и сосредоточился на простых. Тех, с которыми могли справиться джуны почти без помощи новоявленного наставника. Именно почти – определённый процент новизны и сложности Вася сознательно оставил. Чтобы джуны, решая любую задачу, максимально часто сталкивались с чем-то новым.

Сложные задачи Вася решал сам. Поначалу вообще прям сам, но постепенно, месяца через два, начал немного декомпозировать – там, где это возможно. Отдавать куски сложных задач джунам, контролировать выполнение, качество архитектуры и кода.

Результаты, с точки зрения бизнеса, у Васи были отвратительные – в отличие от Коли. Вся команда Васи производила

столько же, сколько он раньше делал один.

Между собой наши герои почти не общались – некогда было.

Осень

Коля очень много психовал, злился, выгорал. Как он ни старался объяснять джунам сложные задачи, поддерживать в процессе решения, улаживать конфликты – проблем становилось всё больше. Доля его участия всё возрастала. Взятые лично на себя задачи он едва успевал выполнять. Нерабочее время, выходные – всё было забито работой.

Времени на общение с заказчиками уходило всё больше с каждым днём. Основной темой разговоров теперь была «Ну Коля, опять твои тут понаделали». Коля, раньше сквозь зубы цедивший «разберусь» и шедший разбираться, теперь говорил «сам доделаю», гневно объявлял джуну «я забираю у тебя задачу» и садился закапываться сам.

Его джуны поначалу переживали – лидер злится, мы что-то делаем не так, надо что-то менять. Однако, понаблюдав за повторяющимся паттерном, сели Коле на шею. Не делали задачу, а делали вид, что делают задачу. Ну, какой-то код писали, что-то проверяли, тестировали, но до результата почти никогда не доходили. Достаточно было дожждаться, пока заказчик взвояет, нажалуется Коле, и всё случится само собой.

Новых джунов Коля не брал – не смог бы решать задачи ещё за кого-то. Старых не выгонял – гордость не позволяла. Но непрестанно ныл, что людишек ему подсунули сильно так

себе – ничего не могут и, главное, не хотят.

У Васи – примерно наоборот. С заказчиками почти не общался – только в самом начале, когда начинался проект, или надо было произвести хорошее впечатление. Модель представления команды была, как у Коли, только ровно наоборот. Вася не говорил «мы профессионалы, решаем сложные задачи». Его девиз – «мы джуны с экспертом за спиной».

К этому времени его джуны уже приближались по квалификации к мидлам. С горем пополам научились и программировать, и с заказчиком разговаривать, и конфликты урегулировать. Не от высокой осознанности и стремления к развитию метапредметных навыков, а потому что иначе не получалось – Вася загребал жар только чужими руками.

Он мог помочь, подсказать, направить, посоветовать – но только своему сотруднику. После первой итерации общения больше с заказчиками не контактировал, хоть кол на голове теши. Никогда не садился дорешивать задачу за джуна, даже если это приводило к потере заказчика.

Результаты команды Васи – как в абсолютном выражении, так и в пересчёте на человека – догнали результаты команды Коли. И продолжали расти, т.к. Вася набирал новых джунов.

Между собой Коля и Вася общались редко. В основном – на совещаниях у начальства, куда обоих приглашали. Вася рассказывал, как у него дела. Коля огрызался.

Зима

Коля ушёл из команды. Или команда ушла из Коли. Мо-

жет, Коля их бросил. Не сошлись характерами. Разные мнения и определения звучат – смотря кого спросить. Но факт один: команды больше нет.

Есть Коля, который снова работает один, решает сложные задачи, нравится заказчикам, менеджерам, руководителям проектов и начальству. И есть несколько джунов, которые за год ничему особо не научились, после расставания с Колей потыкались, помыкались, никуда не приткнулись, как-то самоорганизовались в подобие команды, пересели на несложные задачи и как-то пытаются выжить.

Коля, если спросить, говорит одно: я дал им шанс поработать с экспертом, научиться, подняться и тоже стать уважаемыми людьми. Не захотели, не воспользовались, не подхватили. Я, Коля, сделал всё, что мог. Любой, кто это слушает, понимающие и сочувственно кивает головой.

Джуны Васи стали мидлами, и тащат на себе новых джунов, коих становится всё больше. Даже какие-то тимлидки выросли – пока маленькие, бестолковые, но с 2-3 джунами справляются, и в плане роста навыков, и в части управления с командообразованием.

Дошло дело и до сложного программирования – того самого, с которого начинал свой путь в компании сам Вася. Только теперь его делают мидлы, которые поголовастее. Сам Вася делает только суперсложные вещи – что-то новое, прорывное. Ровно для того, чтобы передать полученный опыт и паттерны мидлам, те – джунам и т.д.

Коля и Вася теперь часто общаются – снова подружились, нет яблока раздора, повода для сравнения, в т.ч. на совещаниях у начальства. Тем более, Коля туда больше не ходит.

P.S.

Я, если честно, не знаю и не берусь судить, кто из друзей прав. К обоим отношусь очень хорошо. У обоих учусь, с обоими нравится взаимодействовать и что-то вместе делать, строить, развивать. Каждый, на текущий момент, хорош в своём.

А вы как думаете?

Леньхаки. Так лень, что не лень

Миша – удивительный чувак. Поразительный и заразителный. Лучший руководитель в региональной IT-компании. Ещё и офигенный программист.

Мы разговаривали всего полчаса, а он успел меня заразить. Вот вы сейчас о чём подумали? Ну, явно не о насморке... Наверное, что он увлечённо рассказывает о повышении эффективности? Новых технологиях? Цифровизации? И я прям такой наслушался и кинулся статью писать?

Нет. Миша – самый ленивый руководитель на свете. Всё, что он любит – программировать. Остальное ему лень. На столько лень, что он готов на всё, лишь бы побыстрее вернуться к программированию. Похоже, потому и эффективный, без кавычек. И его лень заразительна. Так заразительна, что мне лень писать много букв. В кои-то веки буду краток.

Отчётность

Когда Миша стал руководителем, ему велели регулярно заполнять эксельки с показателями – еженедельные, ежемесячные, ежеквартальные, еже-ещё-чего-то-там. Миша раз сделал, два, лень пересилила... Топнул ногой, получил доступ к корпоративной системе, автоматизировал формирование этих показателей. Как ни банально, данные в ней уже были.

Юля, которая требовала эксельку, пожалала плечами и ста-

ла сама брать данные из системы. Миша продолжил программировать. Когда его попросили вывести те же показатели для других отделов, он высоко поднял брови: «а что, в других отделах нет программистов?». Матернулись и отстали.

Совещания

Экселька экселькой, а ещё надо было ходить на совещания. Рассказывать о планах, фактах, проблемах, мероприятиях, корректировках, отвечать на вопросы, срать... Мише было лень. Не сам процесс, а отрываться от программирования, вставать, куда-то переться. Решение пришло не сразу, пришлось поступить, как аналитик – понаблюдать.

Миша заметил две фишки совещания. Первая – чем токсичнее человек, тем реже его зовут и меньше спрашивают. Но есть риск нарваться на неприятности. Вторая – чем результативнее отдел, тем реже человека зовут (не хотят отвлекать), меньше спрашивают (чтобы не обидеть) и... Меньше внимания обращают на его токсичность.

Соединил одно с другим, приложил усилия, отдел стал более результативным. Ну а токсичность пёрла сама – ведь на совещание Миша приходил уже заряженным. Перестали звать вообще. Настолько перестали, что Миша теперь сам, без приглашения, иногда ходит, чтобы хоть как-то понимать, что на фирме делается. Где-то раз в полгода ему достаточно.

Общение с клиентами

Мише всегда лень было разговаривать с клиентами. Вот

прям эмоциональный клинч от одной мысли, особенно если речь о телефоне или видеосвязи. Раз поговорил, два... И нашёл себе «говорилку». Менеджера. Вообще, в компании менеджеров много, они люди важные и нужные. Но разговаривают с клиентами, в основном, «на лидах» - при обращении. Дальше, в деталях, задачу или проект должен обсуждать программист.

Миша предложил менеджеру – а давай все, вообще все разговоры с клиентом будешь вести ты. А я тебе взамен – спецусловия. Ну, во-первых, лично буду работать с твоими клиентами (это уже спецусловие, Миша и правда крут). Во-вторых, сам буду распределять задачи и искать исполнителей (раньше это была головная боль менеджера). В-третьих, лично у тебя будет приоритет перед остальными менеджерами. Получил согласие.

С тех пор Миша разговаривает с клиентами, только когда сам того захочет – да, иногда и такое бывает. Говорит, просто убедиться, что они существуют. А менеджер за год стал лучшим в компании.

Общение с менеджерами

В этом леньхаке Миша чутка слукавил. Проблемой были не только разговоры с клиентами, но и работа с несколькими менеджерами – по правилам компании он, как начальник команды программистов, должен был «обслуживать» несколько менеджеров. Но Мише было так лень... Не программировать, нет.

Лень было выстраивать отношения и правила работы с кучей незнакомых людей. Если помножить их количество на тьму тараканов в головах этих менеджеров, задача получалась нерешаемой. Поэтому Миша, создав эксклюзивные условия для одного менеджера, избавился и от разговоров с клиентами, и от всех остальных менеджеров.

Собеседования

Отделу Миши надо было расти, требовалось набирать людей. Кандидатов поставляли эйчары, собеседования доставались Мише. Проводить их было ужасно лень.

Раз провёл, два, три... Взял людей, посмотрел, как они учатся и потом работают, перечитал ещё раз Даниэля Канемана, понял, что человек на собесе и человек в работе – два совершенно разных персонажа, и надо лишь придумать простые критерии. Придумал.

Раз проверил, два... И выработал простую технологию собеса и адаптации. Научил пару своих программистов всему процессу, и полностью самоустранился.

Зарплата программистов

В компании, когда пришёл Миша, платили ниже рынка – ну, так уж вышло. Программисты, естественно, время от времени затевали душевные разговоры на тему «мало платят, уйду я». Миша раз послушал, два, попытался потушить пожар, о чём-то индивидуально договориться, понял, что ему ужасно лень отвлекаться от программирования на эту несусветную чушь.

Пошёл к начальству, топнул ножкой, потребовал увеличить зарплату в полтора раза. Оказалось, и так собирались, только ждали кого-нибудь, кто ножкой топнет.

Больше этот вопрос Мишу не отвлекал.

Обучение программистов

В компании была отличная система подготовки программистов. Это, короче... Здоровенный файл со ссылками на хрен пойми какие источники в интернете, где надо было чего-то почитать, посмотреть и потыкаться. После – всё, программист!

Миша раз эту интернет-ахиною из головы программиста повычищал, два, за третьего за ночь сел переделал, с четвёртым спорил полдня... И такая лень накатила... Что Миша плюнул и сделал сам.

Сел, упёрся, и изготовил учебные материалы сам. Прямо все, по всем разделам. Потратил несколько выходных, но больше к вопросу не возвращался.

Новые клиенты

Чем больше становилось у Миши программистов, тем больше требовалось клиентов и проектов. Любимый и единственный менеджер, с которым наш герой работал, конечно, старался, но никак не мог нормально продать без участия толкового спеца. А Миша, как вы помните, ни с кем не хотел разговаривать.

Но делать-то что-то надо, а надежды ни на кого нет. Миши, скрипя зубами, раз съездил на встречу, два... Всё класс-

но – продажа состоялась, но, блин, каждый раз по полдня тратить! Мише лень.

Думал он, думал, и придумал. Велел менеджеру конференции организовать. Давай, говорит, собери всех сразу, я приду и всё им объясню – каждый раз ведь одно и то же надо рассказывать. Менеджер, как вы помните, уже был лучшим – а значит, ему компания верила. Надо конференцию – пожалуйста.

Собрали несколько десятков потенциальных клиентов, привезли Мишу, он пару часов разглагольствовал, и вуаля – с десятков новых клиентов. Где-то год можно ни с кем не разговаривать.

Развлечения

Самым страшным открытием для Миши было то, что новоявленным программистам для счастья недостаточно программирования. На то, чтобы с этим смириться, ушёл не один год.

Компания искренне старалась как-то развлекать программистов. Но беда в том, что тамадой всегда был непрограммист – какой-нибудь менеджер, офис-менеджер, продавец и тому подобный Активный Человек, коих среди программистов днём с огнём.

Каждый из них задалбывал Мишу – ну расскажи, что интересно программистам, как организовать, что делать. Миша талдычил одно и то же: программистам интересно программировать. Увы, бывшие профорги и спорторги этого не по-

нимали. Поэтому программистам приходилось играть в настольный теннис, собирать баллисты и кидать мышки, кто дальше.

Миша смотрел, думал и бесился. Наконец не выдержал – это ж просто невозможно, каждый раз отвлекаться от программирования, и получать в обмен на свою жертву такую дичь. Думал-думал и придумал – стал раз в неделю проводить интенсивы по программированию.

Не сказать, что прям сильно с выдумкой подошёл, но результат превзошёл все ожидания. Просто взял программирование и разбавил его традициями корпоративных развлечений – чем-нибудь непреодолимым, необъяснимым, но весёлым.

Тут мне уже совсем лень писать, я приведу цитату самого Миши, извините.

Было примерно так. Участвовали 12 человек, я их разбил на 6 команд по 2 человека.

Поставили 6 компов, везде развёрнута одна и та же исходная среда. Случайным образом рассадили команды по компьютерам.

Озвучил задачу, дал 5 минут. Программирует только один человек из команды, второй стоит рядом, может подсказывать, не может трогать комп.

Через 5 минут все встают, опять случайная рассадка, смена внутри команд: кто стоял - садится, кто сидел - встаёт. Задание - в продолжение предыдущего.

И так несколько раз. 5 минут, переход к случайному компу, смена «программирующего».

Задачи – «из жизни». То одно сделать, то переделать, то всё стереть, то всё вернуть. Что важно - качество решения каждой последующей задачи зависит от предыдущих. А кто там до тебя решал, что и как написал - пойди разберись...

Да, ещё ж соревнование было. Победителем будет тот, у кого окажется лучшее, самое качественное, написанное по всем правилам решение. Разумеется, речь обо всём коде, а не только о последней 5-минутке.

А количество задач заранее неизвестно. Т.е. сидишь ты, делаешь задачу, а она может оказаться последней.

Соответственно, у каждой команды всегда был один и тот же выбор: сделать только «свою часть», опираясь на непонятный код «до меня» (эскалация говнокода, короче), или, по возможности, всё переделать нормально. Заодно была возможность оценить, как это классно, когда до тебя нормально написали и не надо ничего переделывать.

Как в жизни, короче.

Да, больше тема развлечений не поднимается. Всем весело.

А Миша – дальше программирует.

След в туалете

Очень давно я занимался проектами внедрения ЕРП-систем, в роли техлида (тогда это называлось «технический руководитель проекта»). В Челябинской области много производственных предприятий, и часто приходилось ездить на переговоры в сопровождении очень опытной женщины-менеджера.

Однажды она сказала: приходя на предприятие, надо первым делом идти в туалет. Именно по туалету можно очень многое понять о руководстве и его отношении к бизнесу. В первую очередь – о том, мыслит ли оно долгосрочными перспективами, видит ли себя тут надолго, или просто рубит бабло по-быстрому. В те, ещё немножко бешеные годы, пожалуй, доля истины в «туалетном знакомстве» была.

Ритуал соблюдался строго. Менеджер приезжала на встречу и говорила, что сначала надо «припудрить носик». Посещала туалет, осматривала, формировала первичное мнение и шла на переговоры.

Недавно, совершенно случайно, я вспомнил об этой традиции. Старый друг пригласил меня посетить IT-компанию, в которой работает. Заявленная цель – помочь понять, что у них не так с развитием и стратегией. Перед выездом к нему я полез долить омывайки, и чутка изгваздался. Приехав, сначала посетил туалет.

И он меня поразил.

Туалет

Там, конечно, был не космос, роботы задницу не подтирали, но в целом – очень уютно и практично.

Первое, что бросилось в глаза – несколько способов высушить руки. Тут тебе и сушилка, и диспенсер с полотенцами, и бумажные полотенца в рулонах. Выбирай, что душе угодно.

Рядом с умывальником – этажерка, все полки которой уставлены всевозможными полезностями. Крем для рук, одноразовые носовые платки, ролики для чистки одежды (вообще никогда их в офисах не видел), губки для обуви, портативная колонка с играющим радио, и т.д.

И это, акцентирую внимание – в мужском туалете. Который, по традиции, должен быть шедевром минимализма.

В кабинках – идеальная чистота. Дело было днём – явно убирают не раз в сутки. Туалетной бумаги в избытке, двух видов – здоровенный рулон (не знаю, как называется – как в аэропортах) и парочка маленьких, уютных, домашних трёхслойных.

Если верить теории женщины-менеджера, с развитием в этой компании всё должно быть прекрасно. Чё тогда меня позвали?

Стало жутко интересно. Пошёл слушать.

Концепция развития

Компания существовала на рынке достаточно давно – лет 10. Начиналась с нескольких спецов, теперь выросла до двух

сотен человек. Пока народу было немного, всем развитием занимался директор. Как правило, это было развитие по необходимости – затыкание дыр, вызванных ростом.

Ну там, вроде офис побольше снять, компы купить или обновить, брендбук изготовить, сайт сварганить и т.д. Как возникает необходимость – так что-то и делают. Без долгосрочного стратегического видения, будь оно неладно.

Однако, с ростом компании, директор был вынужден развитие делегировать. У самого времени не хватало – как говорят неэффективные эффективные менеджеры, «операционка съела». Почитал книжки и интернет, посоветовался с друзьями и подчинёнными, пришёл к ожидаемому выводу – надо нанять для развития компании отдельного человека. Директора, или хотя бы менеджера по развитию.

Сказано – сделано. Собственно, рассказ моего друга и был построен вокруг этих менеджеров по развитию и их деятельности. Ибо несколько их было, и каждый след оставил свой.

Первый

Первый менеджер заложил основу – структуру и процессы изменений. Разделил развитие на два принципиальных потока: стратегический и демократический.

Стратегический – это долгосрочные цели и, соответственно, длинные программы развития. То, что нельзя сделать быстро, но обойтись без этого никак нельзя – будущее не простит.

Демократический – это всякие сборы предложений от со-

трудников и их реализация. Честно сказал, что толку от них ноль, но «электорат» должен чувствовать собственные возможности по изменению компании к лучшему. Иначе и в долгосрочку играть откажется, ну или будет тихо саботировать.

Этот же, первый менеджер по развитию, провёл вдумчивый анализ проблем компании, которые надо в итоге победить. Если укрупнённо, то это текучка кадров, плохое (почти никакое) развитие спецов, отвратительные продавцы (при хороших и даже растущих продажах), всеобщее потребительское отношение к компании.

Недолго думая, решили взяться за текучку. Подняли протоколы интервью при увольнении, сразу увидели причину – не самая высокая зарплата. Она не угнетающе низкая, но явно меньше рынка. Программисты приходили в компанию, в основном, чтобы получить стартовый опыт и двигать дальше – туда, где платят больше или хотя бы нормально.

Менеджер по развитию, разумеется, провёл расчёты, согласовал их результаты с финансистами и экономистами, и, вроде как, по всем статьям получалось – зарплату можно весьма существенно повысить. Пришёл с предложением к директору. Что услышал? Я подумаю, а ты пока займись другими изменениями.

Другие изменения были из «демократического» списка. Касались, в основном, мест общего пользования – туалета и кухни. В то время не было даже сушилки для рук. Ну и ко-

фемашины. Сравнив стоимость обоих агрегатов, менеджер выбрал сушилку.

Так туалет стал немного лучше.

С решением по повышению зарплаты директор тянул. Сначала вежливо, потом агрессивно – вроде как «отстань, без этого как-то жили, у нас другая концепция». Менеджер подождал, подождал, сказал «да ну вас нахер», плюнул и ушёл.

Второй

Второй менеджер по развитию тоже хотел начать с анализа проблем – мягко отказали, мол мы уже и так всё знаем. Выдали структуру плана развития – стратегические и демократические списки, и велели заниматься подготовкой спецов. Чтобы стала системная, постоянная, управляемая и осязаемая.

В жизни всё было так. Приходит человек работать в компанию, ему всучивают курс подготовки собственной разработки. Достаточно короткий – на один месяц. В этом курсе рассматриваются основные вопросы работы ПО, внедрением и кастомизацией которого занимается компания. Обращаю внимание – именно вопросы работы, а не доработки и развития. Проще говоря, как им пользоваться.

Спрос и на продукт, и на услуги по его внедрению/сопровождению, был достаточно высокий – и раньше, и сейчас. Этот спрос застилал глаза и компании, и спецам (если их можно так назвать). Формально, люди приходили, что-

бы стать программистами. Но услуги по программированию и консультированию оплачивались одинаково. При высоком спросе именно на консультации практически все спецы не программировали вообще.

В результате, зарплата росла очень быстро – считай, месяц поучился, и уже нормально денег поднимаешь, ибо продукт несложный. Год, два сидишь, и только тогда понимаешь, что есть потолок и как ты в него башкой-то и упёрся. Вроде как самое время программировать начать, но опыт всех предшественников говорил – резко просядешь в деньгах.

Собрав всю эту информацию, менеджер по развитию пошёл к директору и сказал – надо менять систему подготовки. Заодно – и правила распределения задач, чтобы было больше программирования. Надо либо купить готовые курсы, либо разработать свои. Да, придётся существенно снизить доходность от спецов в первые месяцы работы. Но в долгосрочном плане это принесёт очень серьёзные дивиденды.

Что услышал? Я подумаю, а ты займись пока другими изменениями. Так в туалете появился диспенсер с бумажными полотенцами.

Третий

Третьему велели заняться продавцами. Как вы уже поняли, проблем с продажами не было из-за удачного выбора продукта – кому-то когда-то то ли очень повезло, то ли реально провидец сработал. Чтобы продавать продукт и услуги компании, достаточно было уметь говорить.

Проблемы начинались после продажи. Любая конфликтная ситуация вводила менеджера по продажам в полный ступор, с единственно возможным ответом клиенту – «я разберусь и вам перезвоню». Ни в чём разобраться не получалось. Перезванивать было страшно.

Результат был неутешителен. Во-первых, среди менеджеров по продажам текучка была выше, чем среди спецов. Средний срок службы менеджера – менее полугода. Во-вторых, убого низкая сохранность клиентов. Продать продукт получалось хорошо. Внедрить, сопровождать, развивать – отвратительно. Порог в «один год сотрудничества» переходили единицы процентов клиентов.

Как вы уже поняли, подготовка менеджеров по продажам была равна нулю. По ощущениям, на эти должности просто сгребали всех подряд, на соседней остановке общественного транспорта или бирже труда. Возможно, учили презентовать продукт. Но никак не объясняли принципы оказания услуг по программированию.

Третий менеджер по развитию, естественно, пришёл с программой, как «подружить» программистов с менеджерами. Требовались и системные, и структурные перестановки в отделах и процессах. Тут не надо быть семи пядей во лбу, чтобы понять – после насыщения рынка продуктом компании ждут серьёзные проблемы. Рынок ведь растёт медленнее.

Что сказал директору этому менеджеру по развитию? Я

подумаю, а ты займись пока другими изменениями. Так в туалете появилась этажерка с бумажными полотенцами, кремом для рук, роликами для чистки одежды и проч.

Сейчас

Сейчас в компании работает четвёртый менеджер по развитию. Как я понял, занимается поиском путей преодоления всеобщего потребительского отношения к компании.

Даже интересно, к каким выводам он придёт?

И что появится в туалете на этот раз?

Клуб использованных техлидов

Меня зовут Сергей. Я – использованный техлид. Долго думал, что я такой один – невезучий, потерявший всякий интерес к любимому делу, работающий от звонка до звонка. Оказалось, нет.

Нас таких много. Начало, первые счастливые годы, безумный интерес к своему делу – у всех техлидов похожи. Середина – пик, душевный подъём, работа без усталости сутки напролёт – один в один. Послушаешь любого техлида про его золотые годы, и скажешь – «во-во, у меня так же было!». Красная нить тех счастливых времён – невероятная востребованность, от которой буквально кружится голова.

Чем больше сложных задач решаешь, тем больше их становится. Чем качественнее, оптимальнее, эффективнее создаёшь, тем больше новых вызовов получаешь. Чем лучше помогаешь другим программистам – молодым, но уже увлечённым – тем сильнее хочется жить, работать, развиваться.

Ну а потом... Приходит умный эффективный дядя, который говорит: экой тут у нас талантище! Да зазря погибает в своём пыльном углу! Не годится. Не эффективно. Надо его использовать.

Сергей

Сергей – это я. Несколько лет назад увлёкся темой использованных техлидов, и начал собирать их грустные истории.

Во многом они повторяются – это не проблема техлидов, нет. Это у эффективных дядей фантазия небогатая, на тему использования лучших и влюблённых в своё дело.

Эти истории я и хочу вам рассказать. Если вы – техлид, эксперт, или просто безумно тащитесь от программирования своими руками – возможно, узнаете себя. Вас или уже использовали, или пытаются (искренний респект, если отбиваетесь!), ну или всё ещё впереди.

Начну со своей истории – она и банальна, и чаще всего повторяется.

Я стал техлидом, побыл техлидом, и меня решили сделать то ли тимлидом, то ли менеджером, то ли начальником, то ли РП. Короче – тем, кто «использует и передаёт свой опыт». Прицепили ко мне программистов. Вообще, они и так сидели со мной в одном помещении. Но если раньше мы были друзьями, коллегами, единомышленниками, то теперь стали... Не знаю даже, как это одним называется, класс этот, или тип. Я – начальник, они – подчинённые.

Дальше, думаю, вам всё понятно. Вместо программирования, архитектуры и производительности я стал заниматься отчётами, собраниями и показателями. Технический язык сменился на менеджерский. Не по любви, а по необходимости – технические термины господам, посещающим собрания, не очень понятен.

Отношения с программистами ухудшились, почти до нуля. Я так и не придумал, как выстроить баланс – и за душу о

программировании, и за шкуру о показателях.

Я стал никем – ни техлидом, ни менеджером. Лучшие ушли. Остались те, кто «просто работает». Ну и я теперь такой.

Кирилл

Здравствуйте, я Кирилл. Использованный техлид.

Моя история похожа на предыдущую, только началась по-радужнее. Меня не поставили менеджером, а как бы «прицепили», что ли, к команде программистов. Ребята были опытные, крепкие мидлы, не хватало лишь базового, системообразующего элемента – техлида, чтобы развиваться дальше.

Я не командовал, не ходил на собрания, не считал показатели. Парни программировали, я программировал, вместе что-то придумывали, делали, делились опытом, ошибками, шутили, смеялись, ковырялись, засиживались допоздна.

Чисто для кадрового учёта меня назначили руководителем этой команды. По-ихнему – структурного подразделения. Это меня и сгубило.

В компанию пришёл эдакий... Проныра. То ли полудиректор, то ли зам по развитию – я так и не узнал его роль. Посмотрел на список подразделений, руководителей, глянул показатели и динамику, и пошёл со всеми разговаривать.

Я был честен и дружелюбен. Постарался максимально доходчиво и вежливо объяснить, чем мы тут занимаемся – разработка, рефакторинг, хайлоад и проч. Нам подкидывают задачи – как правило, нерешаемые другими отделами – а мы их

с удовольствием размазываем по стенке. Зарплатой довольны, офис нравится, счастье и гармония. Он покивал, позеркал, поулыбался и ушёл.

Как я узнал позже, этот человек придумал и протолкнул нарратив – меня надо развивать, но не стандартными методами, ибо что-то не так с моей мотивацией. Не получится просто выдать мне цель и ждать, что я за ней побегу. Я, мол, хочу быть лучшим – и это надо использовать. Конкуренцию мне создать.

Ну и создал. Организовал параллельный отдел. Вложил кучу ресурсов, но набрал несколько крепких отморозенных мидлов. И весьма эффективного тимлида к ним. Зарплату им тупо влупили раза в полтора больше, чем у нас – иначе, думаю, не собрали бы.

Тимлид тот был «правильный» - его больше интересовало не качество кода (хотя он был программистом), а показатели и деньги. Ту же религию он насаждал в своём отделе. А затеявший всё это полудиректор – во всей компании.

Меня начали приглашать на совещания. Просто послушать. Кого? Разумеется, невероятно успешного тимлида. Который сумел красиво сочетать и «высокое» (программирование), и «приземлённое» - деньги, рост показателей, расширение отдела. Я раз послушал, два, три – выдержки мне хватило, даже улыбался, весьма вежливо.

Но на четвёртый раз спросили. Вроде как «чё скажешь, Кирилл?». Я – спокойно, уверенно – мол, ничего против не

имею, пусть ребята растут и развиваются так, как считают нужным. Если есть вопросы ко мне, или нашей команде – с радостью отвечу. Не припомню, чтобы мы кому-то доставляли какие-то проблемы. Промолчали.

Но снежный ком уже было не остановить. То ли я дурак, то ли они слишком умные, тонкие и все из себя психологи, но через пару месяцев я уже ходил на совещания, рисовал графики, и чуть ли не бился об заклад на следующий месяц, кто больше денег заработает.

Теперь я себя ненавижу. Не жалею, нет, не прошу меня оправдать и сам не пытаюсь этого сделать. Сiju, смиренно жду подходящей вакансии и повода для увольнения. Эти твари, я прошу прощения, сумели обронить и взрастить в моей душе зерно сомнения – а техлид ли я? Наверное, это часть психологической обработки – чтобы сбежать не пытался. Ещё и обязанным им себя чувствовать.

Не повторяйте моих ошибок, пожалуйста.

Иван

Здравствуйте, я Иван. Использованный техлид.

Я всегда был одиночкой, никем не управлял и не был управляем. Получал удовольствие от работы и приносил пользу компании. Всё было хорошо, но кто-то там, наверху, решил расширить бизнес. Естественно, за счёт увеличения штата программистов.

Потыкались, помыкались – не могут набрать нужного количества мидлов. Дорого, а переплачивать рынок кому охо-

та? Ну т.е. кому сильно надо – например, рынок захватить – тот ставит коэффициент 1.5-2 и набирает. У наших не было таких ресурсов (или жаба ква-ква), поэтому набрали джунов и полуджунов.

Те, конечно, ни в зуб ногой. Что делать? Ой, у нас же есть Ваня, наш большой и могучий техлид. Пусть подучит, понаставничает, наставит на путь истинный. Зря такой опыт пропадает – надо использовать.

В принципе, я не против опытом делиться – это даже интересно. Сам иногда чуть-чуть переживал, что столько знаний и навыков остаются лишь в моей голове. Согласился я, в общем. Наверное, ещё и потому, что в начале моего пути программиста был такой техлид – очень многому меня научил. Концептуальным вещам. Как он говорил, главное – скелет знаний правильный, а мясом сам обрастёшь.

Но не тут-то было. Джуны перенимали мой опыт очень избирательно. Очень живо интересовались темами «как», но откровенно зевали на «почему», «зачем», «что будет если» и т.п. С какой бы стороны я ни заходил, какой бы формат не пробовал, результат один – ничего, выходящего за рамки текущей задачи, джуны не воспринимают.

А вот когда у любого такого молодца возникает проблема с задачей – тут уж, конечно, «Ваня, приди, понаставничай, объясни мне, что да как». Объясняю, рассказываю, показываю – что было до кочки, о которую джун споткнулся, почему она вообще возникла, как их избегать, и кочка ли это...

Зевает. Как только перехожу к инструкции, что конкретно сделать – оживляется, даже в блокнот что-то записывает. И зачем-то спрашивает меня самого всё поправить – ну, раз уж пришёл. Срочно же, важно и вообще.

Дальше – хуже. Поначалу руководство ещё смотрело на джунов, как на эксперимент – не было ни больших ожиданий, ни великих результатов. Однако, с моей помощью эти ребята стали производить всё больше результата. Моя помощь, естественно, была именно в производстве, а не в обучении и развитии навыков. Я пытался эту мысль донести до начальства, но не сумел. Сказали, что я зря переживаю, всё придёт, надо продолжать делиться опытом.

Теперь джуны были не экспериментом, а бизнес-единицей, приносящей деньги. С соответствующим отношением со стороны руководства – э-ге-гей, давай-давай, молодцы, надо ещё расширяться.

Я, в итоге, прекратился из техлида в помогайку. Ещё не потерял надежду, что со временем джуны хоть чему-то научатся, хоть во что-то вникнут – вынужденно, или как – и я, наконец, смогу снова стать тем, кем хочу. Ужасно надоело подтирать сопли и показывать, как делать элементарные вещи.

Самое поганое – руководство по-прежнему считает эксперимент очень удачным. И присматривается к Коле, техлиду из другого департамента. Тоже сидит, страдает программированием в глубоком одиночестве.

Павел

Здравствуйте, я Паша. Использованный техлид.

Меня решили использовать в продажах проектов. Вообще, это распространённая практика – взять с собой техлида, эксперта, чтобы «сверкнуть экспертизой». Собственно, это и не использование в негативной коннотации нашего клуба, мне такие переговоры даже нравились – потенциальным клиентам мой опыт был действительно интересен.

Но с приходом нового коммерческого директора мой опыт стал неинтересен компании. Он поковырял рынок («провёл маркетинговое исследование») и пришёл к выводу, что компания продаёт и делает не те проекты, поэтому не растёт по деньгам.

Раньше мы продавали ровно то, что умеем делать. А деньги, по утверждению комдира, в других проектах – крупных, с высокими рисками, тендерными процедурами и ЛПР, которым ничего не нужно, кроме освоения бюджетов. Руководство компании поддержало корректировку стратегии.

Меня, по инерции, стали таскать на переговоры. Поначалу я просто изнывал от непонимания, зачем я туда приезжаю. Про мой опыт и экспертность, в контексте интересов заказчика, поговорить было не с кем, по причине отсутствия интересов у заказчика. Я просто сидел и молчал.

Однако, долго так продолжаться не могло. Заказчики задавали вопросы, на которые менеджеры во главе с комдиром ответить не могли. Я тоже не мог, или не хотел.

Вопросы же из другого мира – «автоматизируете всё, все подразделения, к обозначенному сроку в согласованный бюджет?». Раньше заказчики тоже так иногда заходили, но разговор быстро уходил в конструктив – цели, декомпозицию, риски, приоритеты. А тут – «просто ответьте на вопрос, детали меня не интересуют». Что ответить?

«Да, конечно, сделаем». Попросту говоря, пришлось врать. Ибо я, как и вся наша компания, понятия не имели, сделаем ли. В том числе потому, что в момент ответа на вопрос даже примерно не представляли, что надо сделать – в этих «важных для компании проектах» всё выяснялось на ходу.

Ну а дальше – как снежный ком. Я вынужденно вру на переговорах. Компания ввязывается в проект и начинает подыхать, буквально тонуть в постоянно меняющихся требованиях, конфликтах, попытках найти хоть кого-то ответственного у заказчика, чтобы элементарно сдать работы.

К несчастью, в чём-то комдир был прав. Бюджеты таких проектов с лихвой перекрывали все риски, трудности, слёзы и сопли программистов. В том числе – мои. Компания реально стала зарабатывать больше денег, расширилась, неплохо закрепилась на рынке подобных проектов.

Правда, много старых, ключевых программистов свалили – не для того пришли в профессию. В том числе – из-за меня. Ведь я «переметнулся». Да, стратегию придумал комдир. Но кто говорил «конечно сделаем!»?

А я остался. Продолжаю кивать головой и ненавидеть себя за это. Особенно потому, что компания меня очень ценит. Но уже не как техлида.

Снова Сергей

Это снова я, использованный техлид Сергей.

Хочу кратко обратиться к руководителям, которые хотят использовать техлидов не по назначению. Скажу лично от себя, без ненужных обобщений. Хотя историй использованных техлидов у меня ещё много, и везде поворотным моментом является некий некто, который решил сделать некое нечто.

Вы хотите больше зарабатывать, заколачивать, иметь. Я хочу получать достойное вознаграждение. Почувствуйте разницу.

Я хочу, люблю и могу давать то, что нужно людям. Используйте то, что я даю. Учитесь это использовать. А не учите меня давать то, что вы способны использовать.

Меняйте себя, а не меня.

Реверс-хантинг

Случился тут конфуз. Обычно я не провожу собеседования с кандидатами в свой отдел – есть специально обученный человек. Но всяко бывает – человек приболел. И я, по старинке, попёрся сам.

Кандидат оказался знакомым – очень давно работали вместе, были почти друзьями, но несколько лет особо не общались. Увидев меня, парень, почему-то, сильно сконфузился. Даже побледнел. Я не старая страшная обезьяна, и такую реакцию на свой фейс увидел впервые. Что особенно удивительно – это ведь опытный программист, не впервой собес проходить.

Начал отмазываться, что, собственно, просто поглядеть пришёл, как у нас тут всё устроено, и вообще не собирается менять работу, отпустите, я пойду. Но мне стало жутко интересно, чё ж такое творится-то.

Расспросил. Рассказал. Удивился. Оказывается, парень пришёл на собес, чтобы продать своего начальника. Вообще, у них этот процесс поставлен на поток, и так продолжается несколько лет.

Экспромт

Началось всё давно, когда начальник работал на заводе ИТ-директором, и очень хотел нанять пару хороших программистов. В деревне это сделать не очень просто – и де-

фицит постоянный, и, самое главное, хорошие опытные программисты не любят срываться с насиженных мест.

Стандартный поиск кандидатов результата не приносил – текст вакансии привлекал только студентов. Хантинг через сарафанную соц. сеть HR заканчивался «ладно, буду иметь в виду». И тогда начальнику впервые пришла в голову странная идея – самому пойти на собес.

Договорился с директором, чтобы не ругали, и выложил своё резюме. Написал как есть – ИТ-директор, могу быть хоть начальником, хоть программистом, рассмотрю все варианты. Сразу получил несколько приглашений, пошёл. В каждом месте быстро проскочил первичный собес с HR, и добрался, наконец, до своей цели – начальника ИТ-отдела или ведущего программиста. Как вы поняли – ровно до того человека, которого хотел нанять себе.

Дальше просто. Что бы ни сказал ведущий программист о своей компании, наш начальник аргументированно отвечал «а у нас лучше, вот послушай...». Задачи, проекты, отношение, зарплата, рабочее место, парковка, ДМС, контакт с начальством и т.д. Разговор быстро начинал походить на фарс.

Заканчивался разговор одинаково: начальник устало вздыхал и говорил, что ему надо подумать – не поторопился ли он сменить место работы. Вон оно же как здорово-то всё у нас, оказывается! Жалко только, программиста никак не получается найти. А условия просто шикарные, вот послушай...

Пройдя несколько собеседований в течение месяца, он получил сразу двух кандидатов и, соответственно, возможность выбора.

Масштаб

Сначала тупо масштабировал схему, чтобы получить приток кадров. Искал именно заводских программистов, опытных, но не очень уверенных в себе – такие обычно опасаются идти работать в ИТ-компанию. По-человечески, конечно, надо было как все – опубликовать вакансию, подождать, пиар какой-нибудь устроить. Но наш герой плотно увлёкся своим реверс-хантингом.

Научил своих программистов ходить на собесы и хантить тех, кто собирался хантить их. Везло далеко не всегда – зачастую до цели, т.е. ведущего спеца, цепочка не доходила. Иногда его там вообще не было – HR и сразу директор. В таких случаях парни обещали подумать и не перезванивали.

Но если добирались, то во всей красе расписывали, как сами свалили с завода и теперь у них всё распрекрасно. Боятся, мол, нечего – платят больше, задачи проще, а навыки работы внутри клиента – практически бесценны. Старались ходки не бесплатно – за успешную ходку получали разовую выплату в размере месячной зарплаты.

Инфоповод

Дальше – больше. Начальник узрел проблему в продажах – и менеджеры тупые, и до ЛПР не всегда добираются, и ни «бэ» ни «мэ». Реально успешными были только продажи, в

которых участвовал эксперт, но у менеджеров были большие трудности с организацией таких встреч – разве что клиент сам обратится.

Ну наш герой и пошёл устраивать реверс, только уже ЛПР – директорам и собственникам. Искал вакансии, узнавал о компании – выбирал те, где нет своих ИТ-шников, достаточно высокий оборот и нет грустной истории работы с подрядчиком. Представлялся кандидатом, быстро доходил до директора или, в идеале – до кросса со всеми топами сразу.

Включал харизму и экспертность, и почти по каждой озвученной проблеме предлагал решения. Вообще, на обычных продажах он поступал так же, вся разница в инфоповоде. Когда на лицах людей, считавших себя хантерами, расплывались блаженные улыбки, наш парень начинал дрейфовать в сторону «слушайте, я вот что подумал... а вам точно нужен программист на постоянку?». Дальше шли цифры – штатный спец обойдётся во столько, проблемы будут такие, а мы, подрядчик, можем всё сделать за вот такие деньги. Да, в моменте – дороже. Но вы ведь меня знаете. Уже.

Не всегда, но срабатывало. Был и отложенный эффект – возвращались те, кто «вы, конечно, персонаж интересный, мы подумаем...».

Когда штат опытных спецов был набран, а клиентов стало столько, сколько нужно, наш герой занялся внутренней организацией. Подтянул систему подготовки и развития, регламенты обслуживания клиентов, стандарты разработки, рас-

ширил стек и... Начал продавать себя.

Перестарался

Собственно, тут мы возвращаемся к началу истории. Оказывается, у начальника не всё так хорошо в той компании, где он работает. То ли он сам молодец, то ли людей хороших набрал, то ли рак на горе свистнул, но образовался сильный разрыв между его отделом и остальными. Началась сильная внутренняя конкуренция.

Программисты из других отделов начали проситься к нашему герою, а по правилам внутренний хантинг запрещён. Началось безумие – люди увольнялись, чтобы на следующий день прийти на собес к этому чуду-юду. То же было с менеджерами и клиентами.

Руководство попыталось выровнять ситуацию разными способами. Сначала велели нашему герою научить остальных. Он рассказал про реверс-хантинг, все посмеялись. Остальное послушали – про стандарты, процессы, стек – головой покивали, но менять у себя ничего не стали.

Потом руководство попыталось внедрить практику из сельского хозяйства СССР – шефство и присоединение «лежачих» к успешным. Вплоть до временного переселения программистов из успешного отдела в обычный. Чем кончилось, можно не рассказывать.

Почти без реверса

В итоге, осознав своё незавидное положение, наш герой и его команда решили пойти проверенным путём – продать

себя через вариацию реверс-хантинга. Не поштучно, а весь пакет – команда+система, олицетворение которого – их начальник. Ходят теперь и продают его на собеседах.

Целевая аудитория – другие ИТ-компании. Приходит на собес крутой программист, быстро доходит до высокого руководителя. Нужен именно высокий, не средний – тому чужие системы по барабану (я, к слову, средний). И грамотно объясняет, что стал таким крутым спецом по одной простой причине – оказался в правильной системе с мега-начальником. Который, кстати, подумывает о новой работе...

P.S.

На крайний вариант оставили создание бизнеса, всей командой. Надеюсь, после меня им ещё есть куда пойти.

Пусть программирует Серёжа

У каждого клиента – свои предпочтения. Не только в выборе автомобиля, блюда на обед или корпоративной информационной системы. Клиенты любят выбирать программистов.

Ну, что программисты разные – ежу понятно. Считается, что клиенты предпочитают профессионалов. Мы тоже так думали, и искренне стремились сделать каждого своего программиста этим самым профессионалом.

Однако, несколько клиентов, ставя нам задачи, упорно твердили: пусть программирует Серёжа. Хотя Серёжа – лютейший говнокодер, объект всеобщей жалости и главный поставщик материалов для конференций на тему «Как не надо программировать».

Кто такой Серёжа?

Я буду называть Серёжу Серёжей, потому что он просит называть себя именно так. Не знаю, почему.

Ничего особенного. Отучился в ПТУ (колледже) на программиста, пришёл работать. Попал в отдел сопровождения – где надо «а-а-а-а-быстрее-быстрее-клиент-ждёт». Такой старт во многом и определил стиль программирования Серёжи.

Стратегия Серёжи: «быстро и точно». Кратчайшим путём дойти до решения, максимально точно попадающего в ис-

ходную постановку задачи. Особенно важна точность интерфейса, включая цвета, шрифты, названия и порядок колонок и т.д.

Сами понимаете, такая стратегия – чрезвычайно сложна. Реализовать её можно либо будучи бешеным профи, либо – оставляя за собой лютейший говнокод. Серёже, конечно, был доступен только второй сценарий.

Клиенты Серёжу полюбили. Руководитель отдела сопровождения – тоже. Но, тем не менее, через три года Серёжа ушёл – в отдел разработки. Туда, где сидят «настоящие программисты».

Переход

Вообще, переходы были нормальной практикой. Человек осваивает какие-то навыки в сопровождении, и идёт к серьёзным ребятам, задачам и клиентам. Некоторые так и оставались, правда, в сопровождении.

Но Серёжа, будучи почти звездой в сопровождении, нужен был в программировании. Это было не его решение – начальство так постановило.

И началась дичь. И для Серёжи, и для программистов. Закатывали глаза, заламывали руки, брызгали слюной, бились головой о стену. Но ничего не помогало – говнокод стал частью ДНК Серёжи.

Несчастный очень быстро скатился вниз по показателям. Какую бы задачу ему не дали – ни фига не получалось. Точнее, программирование шло с противнейшим скрипом и за-

тратами колоссального времени остальных сотрудников – приходилось объяснять, ещё раз объяснять, тыкать носом, заставлять переделывать и, в конце концов, забирать задачу и, матерясь, переделывать самим.

Начали проседать показатели всего отдела – не сильно, но всё-таки. Ну и атмосфера, конечно, стала гнетущей. Надо было что-то решать.

Позвали начальство.

Решение

Начальство было мудрое. С одной стороны, виден полный провал Серёжи во всём «сурьёзном». С другой – бывшие клиенты Серёжи изнывали без него. Ну а чё, подумало начальство.

Пусть Серёжа сидит с программистами, но решает задачи своих бывших клиентов. Случился лёгкий спор с начальником отдела сопровождения – ведь «забрали клиента». Но, как выяснилось, забирать-то особо нечего, с уходом Серёжи его клиенты почти перестали приносить деньги.

Ну а остальные программисты будут Серёже помогать в развитии навыков, только уже на его задачах и клиентах. На том и порешили.

Пошло хорошо. Первый месяц Серёжа, к великой радости клиентов и начальства, просто херачил весь накопленный пул задач. Быстро и точно, как умел и любил. Но сработал закон Вселенной – чем больше задач клиента ты решаешь, тем больше их становится.

Серёжа перестал справляться с возрастающим потоком. Выход был очевиден – пусть задачи решают и остальные программисты. Те оказались не против. Парней, если честно, немного завидки брали – пусть и говнокодом, но Серёжа зарабатывал больше. А тут – возможность победить засранца на его же поле.

Но начался жесточайший кризис.

Кризис

Программисты полезли в сокровищницу Серёжи и ахнули... Не, говнокод-то они уже видали, но чтобы столько, да в одном месте, и как-то это всё взаимосвязано! И, самое поганое – работает!

Ситуация получалась весьма странная. Раньше Серёжа решал хорошую задачу и огребал за говнокод, который пытался встроить в нормальный, чёткий, лаконичный и выверенный контекст. Теперь – ровно наоборот. Нужно было понять записки сумасшедшего, и хоть как-то вклинить в них что-нибудь изящное и не позорное.

Для программистов это было в новинку. И бесило жутко. Начались скандалы. Брал программист задачу, смотрел, матерился, звал Серёжу и пытался тыкать его носом. Серёжа к тому моменту уже стал чуть смелее, и отвечал просто: «не хочешь – не делай, сам подшаманю». Раз сказал, два сказал – перестали пытаться воззвать к чувству вины.

Ну ладно, сказали программисты. Щас мы покажем – и Серёже, и его клиентам, как надо программировать. И стали

молча делать решения так, как положено. Но тут, совершенно неожиданно, взвыли клиенты.

Во-первых, им не нравились решения программистов – они были не точны. Программист ведь почти Бог, и лучше знает, что надо клиенту. Поэтому делает близко к требованиям, но с обязательным внесением своей лепты в постановку задачи.

Во-вторых, не понравились суммы, которые программисты выставляли за свою работу. Они были в разы больше серёжиных. Оно и понятно – быстрый говнокод дешёв, нормальный – существенно дороже.

Фраза «Пусть программирует Серёжа» звучала всё чаще и настойчивее.

Но программисты не сдавались – решили объяснить клиентам, как надо на самом деле. Организовали несколько встреч, собрались, поговорили. Рассказали про архитектуру, стоимость поддержки кода, правильное проектирование, сценарии использования, масштабируемость и т.д. Короче, изложили, что есть «правильно».

Клиенты послушали, головой покивали, но честно признались – всё понятно, но ничего не понятно. Давайте на примерах. Программисты взялись объяснить на текущих задачах.

Под руку попался инструмент планирования загрузки оборудования имени Серёжи – клиент попросил добавить в него новую функциональность. Программисты сказали, что

на анализ всего инструмента придётся потратить 10 чел/часов. Но раз такое дело, сделают это бесплатно. Клиент, правда, цифру услышал и запомнил.

После анализа программисты сказали, что весь инструмент надо переделать. От макушки и до пят. И анализ сделать ещё раз – только не кода, а требований, процессов и сценариев. С людьми поговорить, короче. А тут уже не обойтись без привлечения бизнес-аналитика.

Насчитали, в итоге, предварительную стоимость в 120-160 чел/часов. Клиент был в шоке – нахера отдавать столько денег за создание с нуля инструмента, который уже давно работает? Программисты пытались объяснить, что это будет правильно. Говнокод обязательно сломается когда-нибудь, не будет масштабироваться, не натянется на него очередное функциональное требование.

Но всё было тщетно. Клиент перестал слушать программистов. Твердил, как заведённый: «Пусть программирует Серёжа». Особенно с учётом того, что Серёжа оценил текущую доработку в 4 чел/часа.

Но программисты не сдавались. Они пошли к начальству.

Финал

Программисты знали, что начальство, конечно, уважает всякие там «технические подробности», но любит деньги. Поэтому решили надавить именно на доход – сравнить суммы от «хорошего» и «плохого» программирования.

За основу взяли тот самый инструмент планирования, ко-

торый оценивали для клиента. На вторую чашу весов положили свою фантазию на тему «за сколько Серёжа это говнокодище сделал», и добавили известную им стоимость доработок инструмента за последние месяцы.

Итак, хороший код принёс бы компании 160 чел/часов. Плохой, по оценкам программистов – 50 чел/часов. Отправили цифры начальству, попросили о встрече.

Начальство, напомним, было мудрое. Решило выслушать и вторую сторону – попросило Серёжу посчитать, сколько компания заработала на создании и доработках этого конкретного инструмента. Ну и назначило встречу на всех.

Серёжа насчитал 300 чел/часов. И не просто насчитал – принёс документы, которые подтверждали цифры. Чаша весов резко качнулась. Но Серёжа, кроме денег, докинул на неё свои говнокодерские аргументы.

Во-первых, инструмент работал уже через месяц после начала создания. Да, через задницу, но задачи клиента решал. Во-вторых, инструмент продолжал развиваться и выполнять своё предназначение в течение двух лет. Ну да, не развиваться, а «развиваться». В-третьих, ни разу, ни единого разочка, с клиентом не было проблем – ни с инструментом планирования, ни с другими задачами.

Четвёртое добавило начальство – неистовая, слепая лояльность клиента. К Серёже, конечно, но через него – ко всей компании.

Программисты с выводами не согласились, но уже исчер-

пали все аргументы. Раз такое дело – пусть и правда программирует Серёжа.

P.S.

Развитие истории получилось неожиданным. Чутка поостыв, программисты подумали, поговорили, поспорили и... Признали право Серёжи и его говнокода на существование. Решили к нему даже присмотреться. И, по возможности, участвовать – очень, почему-то, захотелось и к себе неистойвой лояльности.

Правда, отношения с клиентами Серёжи уже успели испортить. Поэтому решили так: пусть клиент думает, что программирует Серёжа. А задачи будут решать программисты. Не все подряд, разумеется, и не только их – свои клиенты никуда не делись. Никто не собирался менять веру и уходить в говнокод с головой.

Стали просто разбавлять. Один день красиво программируешь, другой – пускаешься во все тяжкие. Сравнивали дни говнокодерства с мероприятиями вроде Ла Томатины. И, как ни странно, остались довольны.

Во-первых, чутка больше зарабатывали – доработки говнокода требовались часто. Во-вторых, нашли в говнокодерстве профессиональную ценность – лучше понимали, как надо писать код. В-третьих, обнаружили, что и своим клиентам иногда полезно сделать «быстро и точно» - лояльность повышается, а следом – и поток объём задач. В-четвертых, клиенты Серёжи, наконец, сдались и разрешили другим про-

граммистам официально решать их задачи.

Теперь можно услышать не только про Серёжу. Пусть программирует Коля, Петя, Вася. Жить стало интереснее и проще.

Конь остановлен, изба догорела

С виду – мужики как мужики. Ну да, среднего возраста, со слегка притухшим взглядом, без искорки. Но таких много среди пытающихся войти в айти. Внешние признаки в резюме и на собеседовании – как у всех, ничего выдающегося.

Но у этих парней есть тайна. Нет, у них Тайна. Вскрывается, как теперь выяснилось, элементарно – задним умом и я крепок. Но то ли я какой-то не такой, то ли явление действительно из ряда вон... Распознать именно тенденцию смог лишь с третьего раза.

История будет короткой, чай не успеете допить.

Я, по злой воле начальства, занимаюсь подготовкой программистов. Принципиально кандидаты мои делятся на две категории: вчерашние студенты и те, кто «одумался», «решился» и проч. По студентам сказать особо нечего – там всё, как у всех. Пришли, повыделывались день-другой, поняли ценность своих знаний и недоделанных вечерних стартапов и сели учиться массивы сортировать.

Спектр жизненных историй одумавшихся намного богаче. Кто только не попадаетеся – учителя, разнорабочие, продавцы, инженеры, слесари, технологи, банковские работники, мастера из цеха, курьеры, несостоявшиеся ИПшники, сисадмины – короче, почти полный срез нашего многогранного общества.

Средний портрет нашего героя таков. Возраст 30-40 лет. Образование – техническое, бывает неоконченное. Работал тем, кем случайно по молодости устроился, и всё никак не мог уйти – платили неплохо, жена, дети, ипотека. Потом *кто-то* подкинул идею, предложил, настоял – в общем, довольно активно подтолкнул войти в айти. Появились курсы, самостоятельное изучение по видео и, конечно, «жгучее желание».

Под этот же портрет подходят и другие парни среднего возраста, которым не суждено стать героями этого текста. Различия проявляются на испытательном сроке. Там нужно сначала решать учебные задачи, потом – реальные, с постепенно возрастающей сложностью.

Итак, первый симптом – человек садится подальше от наставника. И вообще от людей. Благо, кабинет большой и есть полная свобода в выборе места (если оно никем не занято). Монитором к стенке, чтобы его никто не видел. Не критично – мало ли, стесняется человек.

Второй симптом – очень много общается через смартфон. Вообще, «официальные» коммуникации идут через электронную почту и телегу, которую все, для удобства, ставят на компьютер. А тут – смартфон, много смартфона. Много звонков из коридора, много сообщений, написанных и прочитанных под столом. Не критично – мало ли, дела семейные, или долги по прошлой работе.

Третий симптом – стремление забрать работу домой.

Прям жгучее желание. Оно, к сожалению, жутко выпячивается, потому что парни знают «скинуть на флешку» и «выложить в облако» - т.е. что делать с файлом. Но не знают, как этот файл получить (исходники в смысле). Вынуждены обращаться за помощью. Начинаешь спрашивать, нафига тебе домой – уклоняются, мол дома соображаю лучше, ну и «жгучее желание». Не критично – мало ли, я и сам дома вечером лучше соображаю.

Четвертый симптом – полное отсутствие вопросов. Обычно так не бывает. Конечно, поначалу у всех стажёров глаза круглые и вопрос один – «ничего не понятно». Но это вопрос! Начинаешь уточнять, наводить, подсказывать, и процесс со скрипом запускается – вопросы становятся более реальными и конкретными. А у наших героев нет даже вопроса «ничего не понятно». Хоть допытывайся, хоть нет – в ответ уверенная каша. Нормально, разбираюсь, скоро будет результат. Не критично – мало ли, я тоже люблю сам, своим умом до всего доходить.

Тут вы, наверное, подумали, что парни просто списывают. Например, ищут решение в гугле (хотя, сейчас это уже и не назовёшь «списывают» - скорее аутсорсинг мозга). Или просят знакомых программистов помочь. Это нормально, это бывает, но... Не в таком объёме.

Чуваки умудряются протянуть три месяца. Со всеми вышеперечисленными симптомами, но задачи решаются. С низким, но ожидаемым уровнем качества. Да, решение по-

является к утру, но появляется ведь. Что ж, блин, за нахрен? Если предположить, что списывают, то кто этот дурак? Кто настолько, я не знаю... Влюблён! в этих мужчин среднего возраста, что готов в течение трёх месяцев решать за них задачи? Или всё-таки они сами таким вот странным образом программировать учатся?

Нет, увы, не сами. Вскрывается лайвкодингом (теперь это – обязательный элемент программы первого месяца). До смешного – даёшь ту же задачу, что герой решил ночью. И видишь полный, беспросветный тупняк. Ну и вслед – признание.

Жена. Супруга. Любимая. Драгоценная. Всё – она.

Жена пилила много лет насчёт смены профессии. Жена не давала забыть, когда появлялись дети и ипотека. Жена тыкала носом в доход, который не растёт. Жена показывала на знакомых, кто сумел заставить себя. Жена сгоняла с дивана.

Жена заставила смотреть обучающие видео. После «я вообще ничего не понимаю» жена сидела вечерами и сама изучала эти видео. Жена велела купить курс. Жена не выдержала «да, скоро куплю» и купила сама. Жена параллельно смотрела эти видео и прорешивала задачи. Жена писала «дипломный проект» (извините, про современные курсы без кавычек никак).

Жена научилась программировать. Ещё раз. Жена научилась программировать. Жена отправила на собеседования. Жена, естественно, изучила все статьи на тему «что гово-

ритель на собеседовании». Жена проинструктировала. Жена натаскала решать среднестатистические задачи для собеседования. Жена научила правильным словам, после которых на собеседованиях не предлагают решать задачи.

Жена весь испытательный срок проходила испытания. Жена сидела у телефона и быстро отвечала в мессенджерах. Жена днём переделывала все домашние дела, чтобы вечером программировать то, что принёс с работы муж. Жена ближе к ночи будит мужа и объясняет код. Жена с утра догрызает ногти, переживая, сдал или нет. Жена найдёт слова утешения, когда мужа выгонят. Жена горячо одобрит возвращение на прежнюю работу. Жена замолчит о программировании. На какое-то время.

P.S.

Жена, приходи работать программистом. Все проблемы мотивации и адаптации у тебя уже решены. Всё получится. У всех, кто дошёл – получилось.

Начинающие программисты боятся темноты

Не могу больше. Это не моё. Я не тяну. Всю ночь не спал, ворочался, вставал, курил, читал, пытался смотреть телевизор, снова ложился... Ненавижу.

Всех, кто пишет эти радужные истории «войти в айти». Тех, кому я заплатил за курсы по программированию. «Друзей» и «советчиков», которые сказали: смело бросай свою старую, нелюбимую работу, и иди в программисты. Если бы вы только знали, о чём говорите...

Сегодня я уволюсь. Не могу больше выносить этот стыд, позор, невыносимое бессилие. Я не этому учился. Я хочу программировать, писать код. Но за последний месяц я не накорябал ни одной строчки. Я один в темноте.

Утро

А начиналось всё так радужно! Мне 35 лет, и всю жизнь я работал менеджером по продажам... Не имеет значения, чего именно. Да, это работа, на которую я случайно устроился сразу после института, чтобы были хоть какие-то деньги. Начало получаться – ну так, средненько, без экстремумов.

Зарплата потихоньку росла. Сильно потихоньку. Люди приходили, уходили, а я сидел. Жена, дети, ипотека. Получал свою триаху, а то и сорокет, в хорошие месяцы – пол-

тос, как-то справлялись.

Потом кризис среднего возраста, всё как положено. Пить начал. На детей орать, с женой ругаться. Друзья давно меня, скажем так, обошли. Есть кому завидовать.

Когда немного успокоился, начал думать, искать. Вспомнил ненароком, что учился на инженера. Нашёл примеры успеха – тех, кто в моём возрасте ушёл в программисты, даже без технического образования. Впечатлился, решил попробовать.

Купил курсы – как раз акция была, перед НГ. Мне там прям всё нравилось. И график удобный – после работы, вечером, вебинары, общение с преподавателями, решение задач. Я начал программировать!

И чем дальше, тем больше. Я программировал всё, что дают на курсах. Забрал у ребёнка учебник по информатике, порешал задачи оттуда. Нашёл в интернете всякие задачки для обучения – всё прорешал!

Я никогда столько не программировал! Это так классно. Я чувствовал своё, не знаю... Всемогущество! Реально было время, пусть и не очень много, когда я верил в себя, в то, что изменю свою жизнь!

Было, конечно, сожаление – но радостное, до дрожи. И лишь о том, что не допёр до программирования раньше. Я могу. Умею. Получается. Получается!

Курсы я закончил, если можно так выразиться, с золотым дипломом. За время обучения напрограммировал в ра-

зы больше, чем требовалось.

Пошёл искать работу.

День

Искать долго не пришлось. В нашем городе есть лишь две компании, которые берут на работу таких зелёных, как я. Отправил резюме в первую по алфавиту, пригласили на собеседование. С гордостью показал сертификат о прохождении курса – заметил, что немного поморщились.

Дали пару задач на элементарные алгоритмы – решил за несколько минут. Морщиться перестали. Позвали на испытательный срок. Спросил, как что будет проходить. Дело ведь серьёзное: семья, дети, ипотека. Старая, нелюбимая, но стабильная во всех смыслах работа.

Сказали так, типа без обиняков: курсы это хорошо, это навроде изучения алфавита в иностранных языках. А мы тебя будем учить писать и говорить, уж не обессудь. Сначала буду решать учебные задачи, созданные по мотивам реальных. Потом перейду к «живым» задачам, клиентам, проектам.

Меня вполне устроило. Я согласился. На работе, конечно, не обрадовались, но... И не расстроились. Чем порадовали меня. Написал заявление, отработал две недели, торжественно явился в офис программистов.

И пошло-поехало! Не работа, а просто мечта! Мне выдали пачку задач, с полными, чёткими, понятными инструкциями по всему, что с ними связано – и как среду разработки

развернуть, и где что скачать, и в каких модулях, функциях писать код. Задания чёткие, понятные и, сразу видно, не из пальца высосанные.

Я потом поспрашивал – говорят, эти задачи специально собирают в реальной работе. Крутые прогеры, которые пахнут в продакшн, когда замечают хорошую, чёткую, изолированную по контексту и предметной области задачу – складывают её, после решения, в какой-то там каталог. И в итоге она попадает к таким, как я.

А я эти задачи, после лёгких трудностей адаптации, щёлкал, как орешки. Конечно, там не про сортировку массивов, ограниченную рекурсию или универсальную упаковку в JSON, но не сильно сложнее.

А главное – ничто не давит. Ни сроки, ни клиенты. Я делаю, опытные программисты смотрят, дают советы, показывают неоптимальности, я исправляю. Блестяще.

Вечер

Во втором месяце испытательного срока мне начали давать реальные задачи по проектам. Конечно, не самые критичные, срочные или важные. Но – реальные!

Главное их отличие от учебных – я был первопроходцем! Учебные-то прорешали все, кому не лень – и опытные программисты, и такие же зелёные, как я. В реальных, пусть и несложных задачах, первой звездой, альфой, был я.

Процесс решения был несложный. Какой-нибудь опытный, толковый перец – аналитик, или программист – описы-

вали мне задачу, достаточно подробно, начиная с бизнес-задачи, заканчивая названиями модулей, старых и новых зависимостей, таблиц с исходными данными и т.д. От меня требовалось лишь написать код.

Конечно, было сложно, особенно поначалу. Скорее даже не по техническим вопросам, а потому, что теперь от меня что-то зависит, по-настоящему. И мои коллеги, и руководитель, и клиенты. Коих, по рассказам, бывало очень много на одну мою задачу.

Перетерпев первые трудности, я влился. Я снова программировал. Много, интересно. Особенно мне нравилось, как бы это назвать... Равноправие, наверное. Меня стали звать на совещания, спрашивать сроки, проблемы по задачам, знакомить с ходом проекта. Ну, не прям меня лично, но я тоже слушал, наравне со всеми остальными программистами.

Мне всё было понятно, хоть и с некоторыми трудностями и оговорками. Я не всегда знал или понимал, что было до моего кода, и что будет после него. Моё задание было изолированным, объяснённым, понятным. Я просто программировал.

После небольшого торможения в начале второго месяца, я очень много программировал. Я писал код каждый день. Много, трудно, но интересно.

А потом наступила ночь.

Ночь

На третий месяц всё изменилось. В первый же день ру-

ководитель торжественно объявил: «Теперь началась настоящая работа». И загадочно улыбнулся.

Мне дали задачу. Она оказалась совсем непохожей на предыдущие. Не было имён модулей, бизнес-задачи, входа и выхода. Задача примерно звучала, как «тут где-то проблема, надо найти и устранить».

В первый день я, полный энтузиазма, кинулся... Ну, скажем так, просто кинулся. Уже и не помню, куда. Смотреть, листать, мотать, открывать, переходить, искать. С интересом и амбициями первооткрывателя.

На второй день мне стало немного страшно. Особенно под вечер. Я перестал понимать, или просто забыл, что я вообще ищу. Только собрался пойти и ещё раз спросить, как... Мне дали ещё одну задачу. «Распараллелить вот чё там под этой кнопкой на 2-3 потока, если получится». Если получится!

Вообще, я даже немного обрадовался возможности переключиться с совсем непонятной на ещё не понятно, непонятную ли задачу. Ну, кнопку-то я нашёл. Думал, под ней изолированный код. Ан нет, там вызов. За которым код, который с зависимостями, почему-то жутко непохожими друг на друга.

Третий день прошёл в попытках составить для себя какую-то карту этих зависимостей, переходов, передачи аргументов. Три дня я не писал код, и уже сильно переживал по этому поводу. Решил, что пора попросить помощи руководителя.

Позвал его, всё подготовил, открыл, дождался, он подошёл... И я ляпнул «смотрю и ничего не понимаю». Что именно не понимаешь, спросил он. Ну, говорю, куда код написать, который распараллелит. Он – найди корневой вызов, подходящий для изоляции, и сделай обёртку. И ушёл.

Я погуглил, конечно, но понятнее не стало. Прошло ещё два дня. Ни строчки код так и не написал. Я собрался с духом и снова позвал руководителя.

Сказал, что не могу понять, куда писать код. Он ответил, что это нормально, не надо переживать. Но и бросать не надо. Надо искать. И опять ушёл. Снова день без строчки кода.

Дальше я не буду соблюдать хронологию, потому что плохо её помню. Всё как в одурманенном сне. Череду ужасающих открытий о, скажем так, «реальности работы программиста».

Оказалось, что надо понимать чужой код. Это не опция, а обязательство. Этот код писал не я. И не мой руководитель. Зачастую – вообще кто-то не из нашей команды. Иногда даже не из нашей страны. И понимать чужой код надо больше и чаще, чем писать свой. Потому что «вообще-то, весь нужный код уже написан».

Ещё у чужого кода есть диалекты, стандарты, требования к использованию. Знать их все не обязательно, потому что невозможно, но «надо их уважать». Ибо раз я читаю и хочу использовать их код, а не они мой, то уважаемые люди – они.

Если я каким-то чудом найду, где написать свой код, то

надо выдерживать стиль... Стоп, нет. Надо писать код в объединённом стиле – «наш плюс ихний». Как на футболках Студии Артемия Лебедева, где мат написал английскими буквами. Да, кстати, я так и не нашёл, куда написать свой код.

Выслушал по этому поводу несколько речей. Сначала восторженных, типа «за «написать код» надо бороться» или «написать код» надо ещё заслужить». Потом унылых, вроде «я не понимаю, чего ты не понимаешь» или «ладно, давай я заберу задачу и сделаю сам».

Однажды я набрался смелости и выдал предложение. Пусть, как и в прошлом месяце, более опытный программист в моей задаче укажет, где я должен написать код. Ведь этот подход работал! Ну да, я пока не понимаю, не могу собрать в голове весь контекст и стек вызовов. И код чужой плохо понимаю.

Блин, а я откуда должен уметь его понимать-то?! Я нигде не учился понимать код! На курсах меня натаскивали только программировать, разрабатывать, кодировать! В первый и второй месяц работы – аналогично! Я программист, в конце-то концов, пусть и начинающий! Я хочу писать код!

Знаете, что мне ответили? Опытный программист не может мне помочь. Он не найдёт, куда мне писать код. Ему некогда. Он ищет, где написать свой код. И добавили, что это нормально. Так и должно быть.

Этот дикий, нарастающий с каждым днём стресс, про-

длился в течение всего третьего месяца. Я не написал ни одной строки кода. Ни одной! Я так и не понял, куда мне писать код. Пока я копался, задачи у меня забирали. Выдадут, подождут несколько дней, заберут. Говорят, потому что проект ждать не будет.

У меня очень быстро развился полный букет комплексов, от неполноценности до самозванца. Я вообще перестал понимать, кто я, чем занимаюсь, куда иду, зачем всё это. И главное – я уже не знал, программист ли я. И что такое программирование.

Я погряз в непонятно чём. Чтении чужого кода, понимании контекста, попытках воспроизвести ошибку, анализе производительности, понимании диалектов, режимах отладки. Так не должно быть. Не должно!

Программист пишет код. Не читает, не анализирует, не тестирует, не оптимизирует. Пишет код! Для остального есть другие профессии! А я, я... Не смейтесь, но за два месяца я привык писать код. Я это дело даже полюбил.

Самое непонятное и возмутительное: всем окружающим нормально! Я понаблюдал – реально, полдня читают! Чтобы потом написать три строки! И это они называют программированием! Ну или ладно, не они... И это называется программированием?! Разработкой?

Так и должно быть? У всех? Везде? И у вас? Вот вы честно мне скажите – я один такой? Все прям читают код и понимают его? И понимают, зачем понимают?

Не могу больше. Это не моё. Я не тяну. Всю ночь не спал, ворочался, вставал, курил, читал, пытался смотреть телевизор, снова ложился... Ненавижу.

Всех, кто пишет эти радужные истории «войти в айти». Тех, кому я заплатил за курсы по программированию. «Друзей» и «советчиков», которые сказали: смело бросай свою старую, нелюбимую работу, и иди в программисты. Если бы вы только знали, о чём говорите...

Сегодня я уволюсь. Не могу больше выносить этот стыд, позор, невыносимое бессилие. Я не этому учился. Я хочу программировать, писать код. Но за последний месяц я не накорябал ни одной строчки.

Я один в темноте.

Заградотряд

Вот вы как программистов удерживаете в компании? Хорошо платите? Создаёте комфортные условия? Интересные задачи и проекты? Перспективы? Обучение?

И как, получается? Сидят, не уходят? А сколько денег и усилий затрачиваете? Стоит оно того? Наверняка стоит. Но можно и по-другому. Дёшево и бесчеловечно.

Например, используя заградотряд.

Модель удержания программистов в одной известной на локальном рынке компании рассказал друг, который сумел оттуда сбежать. Как он это сделал – расскажу в конце.

Компания

Название, конечно, не скажу, как и локацию. Не гос, не иностранная, не филиал. Но весьма известная в городе-миллионнике. Считается крутой – и среди клиентов, и среди программистов. Выполняет достаточно крутые, по меркам города, проекты.

По легенде, там особый подход, собственная методика разработки, выполнения проектов, работы с заказчиком. Эту методику нельзя купить, послушать на семинаре или скачать в даркнете. Её не выдают бывшие сотрудники компании – нельзя, бумаги подписаны.

Компания работает много лет, бренд сильный и устойчивый, доходы и численность сотрудников неуклонно растут.

Но внутри не всё так красиво. Хотя, постараюсь обойтись без оценок – судите сами.

Набор людей

Там именно набор, как целенаправленное мероприятие, ограниченное по сроку. Они не ищут людей круглый год, только в определённое время. Набор, фильтрация, отбор – к началу обучения. Все, независимо от квалификации и опыта, проходят это обучение. Напомню, там – некая уникальная методика.

Недостатка в кандидатах не испытывали никогда. Зарплата – ниже средней по рынку. Людей привлекает бренд и легенда, на которой он держится.

Итак, сначала идёт жатва – вывешивают вакансии и собирают отклики. Первично смотрит HR, потом – кто-то из понимающих в деле. Дальше очные собеседования. Начинается первый этап обработки.

Кандидатам говорят о том, как им повезло. Ну, что с ними вообще разговаривают. Смотрят опыт, достижения и всегда – всегда! – качают головой, цокают языком и приговаривают «ну не знаю... вам у нас, наверное, будет трудно». И добавляют главное: «наверное, вы этих трудностей не выдержите и уйдёте, в самый неподходящий для нас момент».

Естественно, все кандидаты бьют себя в грудь и утверждают, что ни фиги такого не будет. О видеозаписи собеседования их предупреждали.

Через все тернии набирается команда на обучение. Неза-

висимо от опыта – всех туда. Изучать легендарную методику.

Обучение

Обучение очное, в районе месяца, с теорией, практикой и домашними заданиями. Чему именно там учат – не знаю, друг тоже под подпиской.

Обучение условно бесплатное, о чём всех официально информируют. Если трудоустраиваешься и работаешь – платить не надо. Если увольняешься – изволь выложить кругленькую сумму, примерно 2 месячных оклада.

Никто, естественно, и ухом не ведёт – не для того с таким трудом сюда пробирались.

Корни

После обучения всех торжественно трудоустраивают и начинается, собственно, работа. Но границы организованы весьма примечательно – их нет.

Обычно в разработке есть отделы или команды, как-то привязанные к проектам, фреймворкам, технологиям, клиентам. Это нормально, привычно и объяснимо с точки зрения элементарной логики – должны быть границы, периметр ответственности и своя «стая», или «племя», смотря какую книжку читать.

Здесь не так. Каждого разработчика закапывают в максимально широкий спектр проектов. Тут поковырялся, там поконопатил, здесь забубенил. Контекст и канва каждого проекта – только внутри головы РП. Как эта голова не взрывается – понятия не имею, видимо это часть легендарной ме-

тодики.

В итоге каждый программист очень быстро обрастает «корнями», хорошими и плохими. В одном проекте у него красиво решённые задачи, в другом технический долг, в третьем – его задачи в ожидании тестирования, т.к. зависимы от решений других программистов.

Бренд и чувство вины

В компании есть штатный психолог, который, на самом деле, агитпроп. Обычно психолог сидит, скучает и ждёт посетителей, чтобы помочь им решить какие-нибудь проблемы, пополам личные и профессиональные. Этот же персонаж на месте не сидит вообще.

Работа этого психолога – ходить и напоминать. Вот он и ходит, и напоминает. Принципиально – о двух столпах, на которых зиждется политика в области персонала: бренд и чувство вины.

Про бренд – примерно как директор лагеря из фильма «Добро пожаловать, или посторонним вход воспрещён». Смотрите, какой у нас офис. Вспомните, как здорово вас учили. Посмотрите, какую красивую ёлку на Новый Год вам нарядили (покупают очень дорогие ёлочные игрушки из Европы). Послушайте, что о нас говорят конкуренты. Какие мы проекты крутые делаем. И т.д., во всех вариациях.

Ну и тут же, естественным образом, вытекает разговор о сотрудниках. Бренд вон какой, а вы тут – ну, извините, не тянете пока. Этот вечно говнокодит, тот сроки факапит, те

двое ёлочную игрушку дорожную разбили (колхозники позорные), и вообще весь успех компании – не благодаря, а вопреки таким бездарям.

Но в конце всегда – пару капель позитива. Вы, главное, верьте в себя. Мы – такая компания, такая компания, что не дадим вам сгинуть, мы вас вытянем, поднимем, возвысим, научим. Всё впереди.

И так – постоянно, регулярно, настойчиво. Иногда помогают другие агитаторы – например, директор. Частенько проводятся агитсеминары, где под разным соусом подают всё те же два любимых блюда: величие бренда и чувство вины недостойных его сотрудников.

Нагрузка

Важный элемент системы – постоянно растущая нагрузка на сотрудников. После трудоустройства никого не грузят по полной, но всегда и всем постепенно объём задач увеличивают. Не вслед за ростом опыта – подавляющее большинство сотрудников далеко не джуны.

И не в адаптации дело – работа, в общем, людям знакомая. Одна из целей – проверка границы, нагрузочное тестирование, если хотите, или проверка на отказ. Ну и потом пригодится (ниже расскажу).

Попытка увольнения

Осознавая величие бренда, мучаясь чувством вины, изнывая от непомерной нагрузки, почти каждый сотрудник через несколько месяцев принимает решение уволиться. В его

представлениях процесс этот выглядит стандартно – придёт, скажет, его немного поугovarивают и отпускают. Всё-таки, программисты – птицы вольные, собраться в путь – только подпоясаться.

Заградотряд

Заградотряд – это три человека, ключевая обязанность которых – не давать людям уволиться. Всё, что происходило с сотрудником во время работы, было нужно для подготовки аргументационной базы заградотряда. Грубо говоря, чтобы собрать и подшить дело.

Заградотряд очень быстр. Как только человек изъявил желание уволиться – максимум через час отряд уже сидит с ним в переговорке. В составе отряда – директор, психолог и главный РП. Открывают папку и начинают.

Аргументы выдают последовательно, внимательно глядя на реакцию, чтобы вовремя остановиться, когда жертва «поплывёт». Дальше папка закрывается, и «дорабатывают» чисто на словах. Остальные аргументы пригодятся в следующий раз.

Перечислю основные аргументы. Их выдают в разном порядке, руководит последовательностью психолог – знает, на кого что лучше подействует.

Корни – тут слово берёт главный РП. Перечисляет все долги, косяки и, главное, зависимости по проектам – цепочки задач и продуктов, которые теперь придётся экстренно доделывать, вникать, не спать по ночам, рвать задницу – ко-

му-то из коллег. Возможно, кто-то из них не справится и тоже уволится, а то и сорвётся, заболеет. Ну как так можно поступать с коллегами? Ведь, напомним, каждый программист участвует в максимальном количестве проектов.

Директор говорит про бренд, который он столько лет строил. А ты, засранец, его рушишь. Все сотрудники – и часть бренда, и его бенефициары. Он нужен и полезен всем. Он – наш общий хлеб. А ты, блин...

Дальше логично вытекает аргумент «кто тебя после такого на работу возьмёт?». Ведь бренд, напомним, известный и уважаемый. Увидев его название в резюме, обязательно ведь позвонят. Что им сказать? Вот прям твои сопли процитировать?

Психолог достаёт и показывает видео с собеседа, где, напомним, человеку говорили: тебе будет трудно, ты не справишься. А человек бравировал – херня, я смогу, я сильный и упорный, я хочу у вас работать! Видео смотрят молча.

Делают «вызов»: так где ты настоящий? Там, где хотел большего и был готов учиться, развиваться, вкалывать? Или сейчас, в слезах и соплях? Остановись, успокойся, подумай.

Если ничего не помогло, достают «денежный» аргумент – надо ведь вернуть стоимость обучения. Напомним, это 2 оклада при зарплате ниже рынка. Потом отдашь? Извини, беспроцентные ссуды только для сотрудников. Бери по ставке потребительского кредита.

Заканчивается, как правило, стандартным пряником – его

достають в разный момент, в зависимости от хода разговора. Пряник – снижение нагрузки. Если помните, её постоянно увеличивали.

Если человек в процессе успел «поплыть», то снижение нагрузки преподносится как подарок, вдогонку, потому что мы добрые. Естественно, такой подарок тут же подшивается в дело для следующего вызова заградотряда.

Если сотрудника не проняли аргументы, разговор становится нейтральным «обсуждением недовольства». Спрашивают, что задолбало, с чем трудности. Почти всегда оказывается нагрузка. Предлагают снизить. Это не то, чтобы не проблема, но раз надо – значит надо. Придумаем что-нибудь, подвинемся.

У человека катарсис. Возможно, он хотел не только нагрузку уменьшить, но так радуется, что его перестали давить и улучшили условия, что сразу соглашается остаться. В большинстве случаев – снова начинает лепетать про готовность к трудностям, огромное желание работать в такой компании. И вообще, он не очень-то и хотел увольняться. Ну, знаете, просто чот подустал, наверное. И огромное спасибо за понимание! В дело добавляется ещё один листочек.

Итерации

Как вы понимаете, одной встречей с заградотрядом мало кто ограничивается. Но интервал между такими беседами постепенно увеличивается, по понятным причинам – человеку кажется глупым приходить и ныть по тому же самому

поводу. Ведь в прошлый раз, когда расслабился, успел наговорить. Не дай Бог опять видеозапись была.

А нагрузку снова увеличивают. И объём агитпропа не снижается. Теперь, рассказывая про величие бренда и никчёмность сотрудников, психолог исподтишка подмигивает тем, кто уже встречался с заградотрядом.

Когда человек снова срывается, беседа повторяется. Если в прошлый раз не всё из папки успели достать – продолжают с того места, где остановились. Обязательно достают материалы с прошлой встречи – ну, где человеку снизили нагрузку и он радовался.

Если сетует, что нагрузку-то опять повысили – пожимают плечами, мол «в прошлый раз показалось, что мы поняли друг друга, и это был просто эмоциональный выброс». Нет, говорит человек, мне действительно надо снизить нагрузку. Ну ладно, держи, раз это твой уровень.

Естественно, через пару итераций манипуляции перестают действовать, и заградотряд начинает нормально с человеком разговаривать. Обсуждают и зарплату, и перебор проектов, и возможность специализации, и удалёнку (подальше от агитпропа).

Но такие разговоры случаются редко. Во-первых, мало кто доходит до такой итерации (напомню – интервалы между беседами растут). Во-вторых, сотрудник должен быть ну прям сильно крут и нужен. В-третьих, такой человек будет разрушать систему, на которой всё построено.

P.S.

Да, мой друг-то как оттуда сбежал? Однажды не смог заставить себя пойти на работу, остался дома. Чувство вины заставило отправить СМС «я увольняюсь». На звонки и сообщения не отвечал.

Когда успокоился, раскурил про дистанционное увольнение. Написал заявление и отправил по почте. Пришёл лишь забрать трудовую и кружку. Заградотряд с ним встречаться уже не стал – незачем.

Иррационально свободный

Заявление Николая произвело эффект разорвавшейся бомбы. Целую неделю весь офис только о нём и говорит. В курилке, на кухне, в переговорах и высоких кабинетах. Восхищаются, поливают грязью, пытаются анализировать, думают, как бы его уволить. Но все, без исключения, пытаются узнать побольше об этом парне. Понять, как он стал таким.

Чего же такого он там ляпнул?

Самое обычное совещание, с широким кругом приглашённых – программистов, разработчиков, РП, менеджеров, аналитиков, начальников и проч. Обсуждали офисные фетиши – цели, процессы, обязанности, ответственность, провалы. Кто чего должен и обязан. Тут заметили Колю – он сидел в углу и читал книгу. Заметить удалось потому, что Николай – редкий гость на совещаниях. И кто-то из Больших зачем-то спросил: «Коля, а что входит в твои обязанности?».

Коля поднял голову, улыбнулся и ответил: «Чисто по приколу, назовите хоть одну мою обязанность».

Секунд десять все улыбались. Потом задумались. Несколько минут перешёптывались. Никто так ничего и не назвал.

Немного о Коле

Коля – очень хороший программист. Не настолько, чтобы быть Гуру или капризной Звездой и получать какое-то осо-

бое отношение. Но отношение к нему, действительно, особое. Правда, до вышеозначенного совещания никто этого не понимал и не осознавал.

Грубо говоря, Коля делает, что хочет. А чего не хочет – не делает. Чего Коля хочет или не хочет – никто не знает. Единственный способ узнать – спросить у Коли. Предугадать невозможно.

Коля может назначить встречу с клиентом и не прийти на неё. А может взять и без предупреждения нагрянуть к клиенту в офис.

Коля может не брать трубку, когда ему звонит собственник предприятия-клиента. А может позвонить кладовщику и целый час обсуждать, где лучше мыть машину.

Коля совершенно необъяснимо выбирает задачи. Может взяться за длинную, в несколько недель, тихую спокойную разработку. После ему, естественно, несут следующую, аналогичную. А он отказывается и садится решать очень срочную, но больше слесарную задачу – какую-нибудь поломанную БД починить. Прочитав одно письмо с просьбой об анализе проблемы, Коля его удаляет. Следующее письмо, с аналогичной проблемой – вдумчиво читает, аккуратно отвечает, излагает варианты. Когда его просят решить задачу, он может написать ответ в духе «эту задачу не надо решать». Когда его просят «просто подумать», он присылает почти готовый прототип.

Коля может месяцами не появляться на обязательных со-

вещаниях и мероприятиях. А потом раз – и приползти. Иногда – вообще сам, ни с того ни с сего, совершенно вдруг, организует совещания, конференции и выступления.

Все планы, сроки, проекты, бэклоги, дедлайны Коле по барабану. Он может полгода делать задачу, на которую нужно полдня. А может за полдня сделать то, на что отводился месяц.

Единственное, что в Коле надёжно (но не является его обязанностью): если он взялся, то результат будет великолепен.

Как видите, всё описание трудовых подвигов Коли начинается или с «если», или с «когда». Это не пустые слова, для форсу. Это реальность. Собственно, её Коля и имел в виду.

Ситуация очень странная. Все, абсолютно все привыкли, что Колю надо просить. Не упрашивать или унижаться, а «просить» в противовес «заставлять», «приказывать», «скидывать», «передавать» или «назначать». Настолько привыкли, что уже не замечают. В том числе того, что никак не реагируют на отказ или игнор. Никому и в голову не приходит обижаться, спорить или даже задавать вопросы.

Коля никому ничего не должен. Это было настолько естественно и привычно, что никто даже не задумывался. Лишь явное провозглашение Колей собственной свободы вызвало столь живой отклик всего офиса.

Проблема

Но Коля опять отчебучил. Спустя неделю после своего за-

явления он, ни с того ни с сего, вдруг объявил «Час развития, на котором я Всё Объясню». Естественно, припёрлись все до единого.

Начал Коля издалека. Рассказал про свою первую работу. Там всё было стандартно: задачи, сроки, проекты, планы, цели, бюджеты, «ваша программа не работает», недосып, валерьянка, первая седина (года в 23). Выгорание, увольнение, пауза в месяц.

Какие-то книжки (утверждает, что не помнит название и автора). Говорит, эзотерика, психология и поведенческая экономика. Долгие раздумья. Решение и стратегия.

Но сначала – о проблеме.

Коля пришёл к выводу, что главная проблема любого сотрудника любой компании – предсказуемость. Любой человек что-то собой Представляет. У него есть привычки, принципы, стандарты, ценности, статистически значимые оценки результатов (в т.ч., например, попадание в сроки). Если понаблюдать, говорит Коля, за любым сотрудником – программистом, аналитиком, менеджером – вы заметите некий «стиль», что ли. Наклонности, перекосы, предпочтительные варианты действий, стандартные реакции на типовые раздражители, одни и те же способы уйти от проблем, «предсказуемое геройство», ожидаемое самокопание, легко читаемые крючки для манипуляции и т.д.

Обращаясь к уже знакомому начальнику с просьбой, вы всегда с высокой вероятностью можете предсказать его ре-

акцию, ответ. Поэтому можете к ним подготовиться – аргументами, контраргументами, цифрами.

Ставя задачу программисту, вы заранее знаете, обрадуется он или загрузит. Один обрадуется задаче по новой технологии, у другого случится паническая атака.

При конфликте с клиентом менеджер ведёт себя по одному и тому же, своему личному шаблону. Один всегда ошתי-нивается, другой оправдывается, третий обещает «прибить к стенке этих подонков-программистов», четвёртый скажет «нужно собраться и обсудить это отдельно».

Когда близок дедлайн, каждый человек ведёт себя по одному и тому же шаблону. Один паникует и мельтешит, второй сжимает челюсти и не спит ночами, третий бежит увольняться, четвёртый спокойно, рассудительно придумывает, как вывернуться и получить индульгенцию.

Так вот, говорит Коля... Всё это не годится.

Паттерны

Любой паттерн, шаблон, привычное поведение, по мнению Коли, создаёт паразитирующую зависимость. От неё страдают обе стороны – и сотрудник, и его «потребитель» (внешний или внутренний).

Для сотрудника зависимость в том, что он сознательно или подсознательно вынужден всегда действовать так же, как раньше. По сути, свобода выбора есть только в первых итерациях применения (а точнее - становления) профессиональных навыков. Раз выполнил задачу в срок, второй, третий

и... Всё, паттерн сформирован. Теперь путь, ценность, критерий правильности только один – всегда выполнять задачу в срок. Любое отклонение воспринимается и сотрудником, и постановщиком, как провал, ошибка, выход из строя.

Никто не хочет провалов и ошибок, несоответствия ожиданиям, поэтому сотрудник вынужден вести себя ровно так, как в предыдущих итерациях. Быть таким, каким его привыкли видеть. Точнее, каким сам приучил видеть себя.

Для «потребителя», или постановщика, или просителя зависимость в том, что он чувствует себя обязанным укладывать свою задачу и все её параметры в некие известные заранее особенности исполнителя. Программист Серёжа принимает задачи только по электронной почте – ага, надо не ходить, а письмо написать. Вася всегда ошибается при оценке срока в два раза – надо не забыть умножить перед тем, как сказать клиенту. Этот начальник всегда плохо воспринимает просьбы о покупке мебели во второй половине дня – слишком плотно кушает в обед – надо, значит, приехать пораньше и сразу к нему.

Как видите, продолжал Коля, эти ребята только усиливают зависимость. Сотрудник следует собственным паттернам, боясь не соответствовать ожиданиям. Постановщик, укладывая свои требования в этот же паттерн, лишь подтверждает его, паттерна, значимость. И в глазах сотрудника, и в собственных («сделал всё, как он любит, и получилось!»).

Так, по замкнутой петле обратной связи, паттерн стано-

вится железобетонным. Разрывается, увы, только увольнением – в новой компании снова будут первые итерации с «утверждением паттерна». Возможно, удастся сделать его лучше.

Сознательная иррациональность

Итак, если проблема в следовании паттерну, то как её победить? Не следовать паттерну, конечно. Или следовать не паттерну, если угодно. Действовать иррационально и непредсказуемо.

Именно к такому выводу пришёл Коля. Устроился на новую работу, попробовал. Сначала были конфликты, срывы, скандалы. Главный ведь паттерн в том, что у всех есть паттерны. Человек, который ведёт себя иначе, вызывает смешанные чувства – интерес, отторжение, страх, ненависть, зависть и т.д.

Потерпев 2-3 месяца, ведя себя иррационально, Коля стал абсолютно свободен. Никто, включая собственника, не «ставил» ему задачи. Всем ставили, а Коле – нет. Предлагали, просили – да. Но не заставляли.

Коля же вёл себя ровно так, как в нашем офисе, как написано в начале. То брал задачу, то не брал. То просрачивал, то делал в сотни раз быстрее. То отказывался читать запрос, то бегал и приставал с многочасовыми расспросами. И так – абсолютно всё.

Говорит, это очень просто: надо делать то, чего от тебя в данный момент не ожидают. Максимально игнорируя обсто-

ательства, возможный ущерб, обиды и т.д. Конечно, есть какие-то прям совсем берега, где речь идёт о серьёзных потерях в деньгах, но не в нашей, программистской профессии.

Предупредил, что надо правильно понимать фразу «то, чего не ожидают». Ключевая, говорит, ошибка – тупо делать инверсию. Люди не дураки, быстро заметят этот фокус – «послушай и сделай наоборот».

Не стоит устраивать и подобие карусели, т.е. выбирать варианты поведения по очереди. Или привязываться к дням: в понедельник я добрый, во вторник – злой, в среду конструктивный и т.д. Это будет рациональная иррациональность. А иррациональность должна быть иррациональной.

Главное, по словам Коли, пережить первые пару месяцев. Люди будут в бешенстве. Лучший рецепт прививания собственной иррациональности – собственный же профессионализм. Та самая единственная предсказуемость Коли: если взялся, то результат должен быть прекрасен.

Отдельную проблему представляют новые люди – клиенты, менеджеры, начальники и т.д. Все «старые» привыкли, а новые будут возмущаться. Тут рецепта два: или действовать, как в первые месяцы, или посылать «новых» к «старым». Коля предпочитал второе – отправлял нового менеджера к старому с заданием «иди и узнай, как со мной правильно работать».

Тут стоит упомянуть о тех, кто с Колей, собственно работает.

Клиенты Коли – самые довольные и стабильные в плане денежных поступлений. Проверяется их «довольство» просто. Если позвонить и спросить мнение о Коле, наговорят гадостей. Если тут же предложить заменить Колю кем угодно, устроят бунт.

Менеджеры и РП, работающие с Колей, аналогичного мнения. Спросишь – скажут «очень трудный человек». Предложить переключиться на других программистов – наотрез откажутся. Спросишь причину – «без комментариев».

Никаких розовых соплей и восторженных отзывов. Но зависимость от Коли предельно жёсткая, необъяснимая, уже на уровне рефлекса.

У Коли же зависимостей никаких.

P.S.

Коля так живёт уже 12 лет. Работа у нас - третья с момента перехода к иррациональности. Менял работу потому, что... Скучно.

Крепостное право в ИТ

Я уж и забыл про крепостных в ИТ, но сегодня обратился знакомый и всколыхнул воспоминания пятилетней давности. Симптомы один в один: деревенский ИТ-директор решил сменить работу, но никак не может найти предложение с зарплатой, даже близко похожей на текущую. Потому что никому не нужны крепостные крестьяне.

Крепостными становятся, обычно, ИТ-специалисты, работающие в штате клиентов. Соответственно, это ИТ-директора, программисты и сисадмины. Есть ли такое в ИТ-компаниях – не знаю, тут интересно вас послушать.

Кратко напишу, что это за дичь.

Краткая история крепостного права

Что там было в царской России – не буду пересказывать, мы с вами одни учебники истории читали. Но в 1861 году крепостное право не исчезло, а модифицировалось – стало, пожалуй, гибридным.

Цель его осталась прежней – «закрепостить», т.е. сделать так, чтобы сотрудник не мог сбежать к другому работодателю. Если помните, с этого и началось крепостное право.

Только методы стали мягче. В основном это очереди на жильё (уволиться – встанешь в конец очереди), беспроцентные займы (или льготные кредиты), условия которых резко плохее при увольнении, ну и главное – «выслуга лет» в раз-

ной форме.

Сталин на какое-то время вообще вводил уголовную ответственность за беготню между заводами. Потом стало мягче, как в предыдущем абзаце. Часть методов переключалась в 90-е (помню беспроцентные кредиты родителей, при отсутствии выплаты зарплаты вообще), потом в нулевые (лично брал беспроцентные займы на первой работе), сейчас этого сильно меньше – остались, например, займы на образование (менеджеры учатся на МВА тыщ за 500, при увольнении займ превращается в так-себе-кредит).

Но появился другой метод, характерный в основном для ИТ. Назовём его «ползучка».

Ползучка

Вообще, его называют «ползучее увеличение зарплаты». Но я бы назвал ползучкой ещё и карьеру будущего крепостного.

Начинается почти всегда одинаково: человек приходит работать специалистом – программистом или сисадмином. Большинство на одном месте долго не задерживаются – получили опыт, применили метод «прыжок» и ушли на другое предприятие, получив плюс чирик к зарплате. Но только не наш абстрактный герой.

Наш держится и проникает, как спрут. Поначалу – в профессиональном плане. Глубоко погружается в ИТ-инфраструктуру, бизнес-процессы, потребности, сценарии, проблемы, ошибки. Вот прям глубоко, как для себя – больше,

чем требуется. Задаёт вопросы, проявляет рвение, радеет за предприятие.

Естественно, лояльность не остаётся незамеченной. Человека постепенно начинают воспринимать, как своего. Как надёжного, который выручит в сложной ситуации. И он выручает. Особенно в кризис, будь то взлом, внезапная проверка, пожар, потеря клиентов, расширение бизнеса и т.д.

Он такой, молодец. Все остальные ждут задачи, пинка, задания, а наш сам знает, что делать. Запускается новый магазин? Наш герой там ночует. Пришла проверка? Вывозит сервер в багажнике домой. Воруют на складе? Быстро внедряет какую-нибудь RFID-маркировку или «предметно-количественный учёт» (что бы это ни значило). Короче, добивается результата в рамках предоставленных ресурсов.

Где-то в процессе поднимает вопрос о повышении зарплаты. Или вопрос поднимается сам. Если заслуги реальные, то вопрос решается позитивно. Ещё раз акцентирую: зарплата поднимается не по какой-нибудь там «объективной шкале рынка», не за профессиональные компетенции или грейды, а потому что молодец. Потому что выручил, помог, участвовал, бежал впереди паровоза.

Раз повысили, два повысили, потом и о должности разговор зашёл. Чего такому человеку программистом числиться? Ведь он все процессы знает вдоль и поперёк, лучше главбуха, директора и финдира вместе взятых. Пусть будет ИТ-директором. А что это значит – сам придумает.

Очень скоро зарплата начинает обгонять рынок. Но это никого не беспокоит – ведь понятно, за что платится. За надёжность. За то, что знает нашу систему и процессы, как свои пять пальцев. За то, что не надо объяснять.

В некоторых случаях человек не просит себе должность, остаётся программистом или сисадмином. Бывает, что просит, но не дают – тоже не дураки сидят, знают, что Большого Начальника потом скovyрнуть сложнее. Некоторые программисты сами отказываются – не хотят брать на себя ответственность за непонятные абстракции вроде «стратегия развития».

Ползучка обычно продолжается несколько лет. Самый длинный известный мне пример – 13 лет. Наверняка есть больше, я сужу по своим знакомым.

После 10+ лет ползучки зарплата у человека в 2-3 раза выше рынка. Вот прям смеяться будете, но в какой-нибудь провинции это могут быть сотни тысяч.

И тут, почти всегда, приходит Тупик.

Тупик

Всё просто – зарплату в разы выше рынка видит не только крепостной, но и тот, кто платит. Вообще, т.к. зарплата растёт не спеша, ползуче, как коммуналка, то особо и не замечается. Но иногда, вдруг, когда вожжа под хвост попадает, собственник или директор сумму-то замечают.

Частенько на правильную мысль наводят доброжелатели – смежники, из других ведомств, вроде снабженцев, про-

давцов, производителей и т.д. У большинства из них есть ключевое преимущество – результаты можно измерить в деньгах, и зарплата – лишь некая «доля». В случае ИТ-специалиста все копыя сломаешь, но хрен поймёшь, за что ему платят. За то, что «всё работает».

Разговоры на тему «а за что мы столько платим?!» возникают всё чаще. К этому моменту, разумеется, наш крепостной (ещё не понявший своей зависимости) уже здорово научился проводить разговоры о своей зарплате грамотно. Тут голову потупит, там психанёт, а то и заявление на стол кинет, или уведёт разговор в песок – предложит посчитать стоимость обслуживания внешней компанией («пусть только они сначала разберутся, как тут всё устроено»).

Разговоры, инициированные текущим руководством, почти всегда заканчиваются ничем. Поорали, пропсиховались, извинились и дальше работать.

Но у тупика есть и обратная сторона: крепостному всё сложнее осуществлять ползучее увеличение зарплаты. Он настолько хорошо всё организовал, что уже сложно придумать, почему ему надо больше платить. Доходит до смешного – чувак сам создаёт кризис (например, бездействием). Или с нетерпением ждёт объективной необходимости в себе – например, строительства нового цеха, появления нового вида деятельности, или выхода на IPO.

Однако, всё равно наступает момент, когда единственным гарантированным способом повышения зарплаты, уже и так

в 2-3 раза превышающей рынок, становится каприз. Надо ходить грустным, хмурым, задумчивым, выражать недовольство и ждать, пока вызовут и спросят, что случилось. Тут выдаётся заготовленная история о... Чём-нибудь, связанном с зарплатой. Денег не хватает, перспектив нет, друзья больше зарабатывают (никто не проверит), хочется больших задач и т.д.

Начальство не любит, когда хорошие, надёжные люди грустят. Поэтому добавит денег – «лишь бы не плакало». Но когда-то сказке приходит конец. По моим наблюдениям, сценарий почти всегда один и тот же.

Вот тебе, бабушка, и Юрьев День

Однажды меняется ЛПР. Как правило, появляется новый директор. В системе ценностей собственника директор, естественно, выше любого из деревенского совета директоров. Часто бывает так, что новый директор нанимается именно для того, чтобы принять неудобное для собственника решение – разогнать всю эту плеяду, хотя бы чужими руками.

Новому директору, естественно, наплевать на бывшие заслуги и всякие там ползучки. Он оценивает зарплаты по рынку и результатам. У продавцов, производителей и прочих, напомним, можно понять корреляцию между результатами и получкой. Объяснить зарплату крепостного ИТшника, увы, разумными доводами просто невозможно.

Его не выгоняют сразу. Просто задают массу неудобных вопросов и перестают предсказуемо реагировать на капризы.

Речь уже не идёт о повышении зарплаты – чуваку предлагают обосновать, за что он, собственно, получает такие деньги. Тем более, что в загашнике нового директора почти всегда имеется знакомый – или интегратор, или Вася с прошлой работы (такой же крепостной).

Тут наш трудяга и начинает искать новую работу. Пока – в розовых очках.

Реальность

Что происходит дальше – сами, наверное, догадываетесь. Крепостной выходит на рынок и узнаёт реальную оценку своим... Чему?

Казалось бы, навыкам и компетенциям. Но очень быстро становится понятно, что навыков-то особо и нет. Всё, что человек умеет и знает, относилось к конкретной среде обитания. Наверное, возможно, когда-то, где-то получится эти навыки воспроизвести, показать и доказать. Но собес идёт за собесом, а доказать не получается. Банально нет таких тестов.

Любая проверка навыков программирования заканчивается фразой «ну, у меня там было немного по-другому». Старые технологии, никому (теперь) не понятные приёмчики и фреймворки, масса кода, который «нельзя показать, сами понимаете», но я обязательно разберусь! Аналогично – по сисадминству.

Но самое главное, что пытается продать крепостной, вообще никакой проверки на прочность не выдерживает. Он

ведь чего делал? Процессы, нервную систему бизнеса, железным кулаком за яйца держал (простите, это фраза одного из подобных кандидатов с собеседования, чтобы вы понимали их подходы и лексикон). Но стоит такому яйцедержателю рассказать о наших процессах – начинает тупить, как вчерашний выпускник какого-нибудь новоиспечённого факультета бизнес-анализа.

Тут же оказывается, что все знания о бизнес-процессах жёстко привязаны, как минимум, к отрасли – и это если повезёт. Обычно же, увы, к конкретному предприятию и его персоналиям. Ни знания, ни опыт, ни результаты нельзя абстрагировать, перенести, воспроизвести за обозримое время.

Никаких кейсов крепостной забрать с собой не может. Уходит гол, как сокол. Лишь с могучим «Я могу!».

Соответственно, и на прежнюю зарплату рассчитывать не может. Даже на её половину. Изредка, конечно, случаются чудеса. Например, попадаетесь ЛПР, знакомый с прежним работодателем, а то и служивший там. Или крепостной устраивается в компанию-конкурент, и ему платят больше за информацию, чем за результаты.

В большинстве же случаев, как ни прискорбно, начинать приходится от печки – ровно с той зарплаты, с которой началось крепостное право.

Чё делать?

Чё делать, чё делать... Разумно балансировать. Если есть

возможность, не меняя место работы, повысить зарплату «за лояльность» - почему бы и нет. Но всегда помнить, что придёт «освобождение» – всегда, как бы ни было хорошо сейчас.

Развивать, прокачивать и применять реальные навыки. Следить за трендами (не путать с хайпами). Изучать и применять новое. Накапливать максимально абстрактный, отторгаемый, широко применимый в будущем опыт. Короче, работать на комплект увольнения.

Ну а если уж связались с бизнес-процессами и бизнес-результатами, то добро пожаловать в большой мир Упаковки и Продажи. Делайте это заранее.

PROграммист со звёздочкой*

Про качество подготовки программистов на раскрученных курсах не написал ещё только ленивый. Я – ленивый, поэтому пишу только сейчас.

Попался мне тут человек, который принёс на собеседование диплом, выданный одной раскрученной образовательной организацией. Раньше я только жалобы на неё читал, что деньги не возвращает. Но никогда не видел выпускников. Стало жутко интересно.

Для этого кандидата я пропустил свои стандартные проверки на логику и алгоритмы, и просто взял его на испытательный срок. Не смог удержаться – очень хотелось посмотреть, чему же учат. Пока человек рассказывал, я сбегал в интернет, прочитал и ахнул.

Судя по описанию курса и его результатов, чувак должен меня собеседовать, а не я его. В названии курса красовался эффектный уровень PRO.

Итак, взял я чувака на испытательный срок. Сравнить мне есть с кем – в двух словах опишу этих ребят.

В основном ко мне в отдел приходят люди, которые хотят стать программистами. Обычно входных критериев два – какое-нибудь ИТ-образование (чтобы цикл от условия отличал) и решение пары задач на логику и алгоритмы на собеседовании. Дальше идёт погружение.

Скорее, правда, «погружение». В программировании оно простое донельзя. Есть широко известная в узких кругах книга чудесного автора, которая очень поверхностно знакомит со всеми основными аспектами разработки. Фишка в том, что книгу надо не просто читать, а «прорешивать» - набивать в компьютере код, который напечатан на бумаге. Один большой, на неделю, сквозной пример.

Знаний и умений после книги почти нет. Всё, что она даёт – знакомство со средой разработки, лёгкая тренировка по программированию на русском языке, ну и запомнить несколько сочетаний клавиш, пару-тройку свойств, событий, типов объектов.

На этом – всё. Не, вообще, конечно, там написано намного шире и глубже, но в первую неделю никто этой ширины и глубины понять не в состоянии. В реальности остаются лишь поверхностные знания. Которые, однако, очень нужны при решении первых учебных задач – чтобы не тупить на ровном месте и знать, в каком месте писать код.

В этой точке объекты сравнения сошлись. Человек с модных курсов против чуваков, прочитавших за неделю книжку. Обе группы идут изучать уже вполне конкретные темы и сдавать мне экзамены. В первый месяц это прям вообще база – универсальные коллекции (массивы, структуры, деревья, таблицы), запросы и базовые алгоритмы.

Сухая статистика. «Книжники» в первый месяц сдают 2.5 экзамена. «Курсовик» сдал 0. Среди «книжников» такое то-

же бывало – правда, лишь один раз (оказалось, ему кто-то «помог» на собеседовании).

Краткая характеристика навыков, полученных чуваком на курсах: умеет нажимать, куда сказали. Пока всё, больше ничего не заметил. Из языка запросов там преподавали, судя по ощущениям, только `select` и `from`.

На всякий случай жирным шрифтом напишу: человек хороший. С правильными ценностями, целями, подходами. По формальным критериям его пора выгнать, но я не буду – верну его в исходную точку и дам книгу прочитать.

Он просто поверил рекламе. Там, как водится, было гарантировано трудоустройство. Ну и, разумеется, в конце каждого значимого обещания стояла *.

Выносить какой-то вердикт не буду – кто я такой... Если попадётся ещё один кандидат с этих курсов, попрошу-таки решить задачки на логику и алгоритмы. Не стоит исключать, что мне просто попался какой-то необычный чувак. Но диплом-то ему выдали...

Люди с других, менее известных курсов, тоже попадались – были чуть лучше. Но, увы и ах, через месяц абсолютно выравнивались с «книжниками», которые потратили на «обучение» от силы 500 рублей. Продолжу смотреть, наблюдать, сравнивать. Но повода выкидывать книжку, с которой сам начинал в 2005 году, пока не вижу.

Что вижу, то и программирую

Не знаю, как у вас, в большом мире разработки, а у нас, в автоматизации бизнеса, очень распространён подход «что вижу, то и программирую». Есть более удобоваримое название: «программирование от данных». Однако, чаще всего это называют говнокод. Хотя, тут я не согласен – до говнокода ещё надо немного подтянуть.

Обычно, необходимость в программировании от данных возникает под давлением ряда обстоятельств. Например, «надо срочно» или «вотпрямычас» (процентов 90 задач в автоматизации бизнеса). Также случается «нечего там смотреть и анализировать, денег только содрать хотите» (те же 90%, пожалуй). Сверху накладывается «да точно ничего не поменяется через 10 лет» (а чего ему меняться, 90%!). Увы и ах, пересекаем три по девяносто, и получаем решающий фактор: 90% программистов в бизнесе по-другому просто не умеют.

Поглядим на несколько примеров и их отложенных последствий.

Копипаст-отчёт

Попросили программиста сварганить пару отчётов для производственного предприятия. Первый показывал выпуск продукции, второй – отгрузку и остатки. Отчёты, к сожалению, были для директора.

Кто не делал отчёты для директоров старой закалки, я поясню: там всё должно быть на своих местах. Гибкие возможности и настройки не то, чтобы не нужны... Строжайше запрещены. Всё аккуратно, по линейке, в заранее обозначенных местах. Поменять местами колонки так же страшно, как переставить полки с хлебом в Ашане.

На тот момент предприятие производило две группы продукции. Программисту так и сказали – сначала первая группа, затем вторая. Данные налицо, от них и код заплясал. На беду, формулы вычисления цифр для групп несколько различались. Не так, чтобы сильно, но в рамках «вотпрямычас» программист не сумел уложить всё в одну функцию. Ну и написал две – по одной на каждую группу продукции.

Директор, хоть и старой закалки (а может – потому, что старой закалки), бизнес развивал. В том числе – линейку производимой продукции. Через несколько месяцев появилась ещё одна группа продукции – плюс функция в отчёте. И ещё. И ещё. И так несколько лет.

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.