Ошибки, которые я делал на протяжении двух лет работы junior developer’ом и как их избежать

На твоей первой реальной работе software developer’ом, даже самые различные мелочи могут показаться ужасающими. Окончить вуз и отважиться выйти в реальный мир, это большой шаг на пути к успешной жизни, при котором ты будешь спотыкаться много-много раз, пока не научишься крепко стоять на ногах и быть полностью уверенным в себе.

Конечно, вы можете считать себя профессионалом, но позвольте мне задать вам всего лишь один вопрос: “как много ошибок вы сделали”?

Старт карьеры software developer’ом, это не только лишь начало вашей работы, но и начало становления мастером в своем ремесле. Как и в любом деле, у вас есть все шансы на успех, но также существует вероятность неудач. В данной статье я хочу рассказать вам о всех своих ошибках, которые я сделал за два года работы software developer’ом.

**Получение работы**

Найти себе достойную работу сразу же после окончания универа, это все равно что научиться бегать прежде, чем ты научишься ходить. Главное убедись в том, что ты выбрал правильную работу. В компании, где ты будешь работать, тебя должно все устраивать, а также ты должен понимать, в каком именно направлении ты хочешь двигаться дальше.

**Узнайте себе цену**

Когда я получил первую работу software developer’ом, учась в университете на втором курсе, у меня были серьезные проблемы с финансами. Из-за отсутствия денег, я согласился на первый вариант заплаты, который мне предложили. На тот момент, я думал, что мне просто необходимо было принять это предложение, так как в сравнении с ужасной стипендией, оно было просто великолепно. Как же я ошибался. Я понятия не имел, что сумма, которую мне предложили, была очень и очень незначительной.

В скором времени, это ошибка повторилась. За несколько месяцев до окончания универа, я устроился работать на новом месте. На этой работе, мне предложили зарплату на 25% больше, чем на первом рабочем месте.

Как оказалось, зарплату которую мне платили, также была очень небольшой, для сферы в которой я работал. Получается, что мне принижали зарплату, но я, не зная этого, был очень рад работать за нее. На тот момент, я все еще не знал, что не только работодатель способен назначать условия.

Ты также можешь задавать свои условия. Если бы я хотя бы чуть-чуть потратил время на то, чтобы узнать сколько люди зарабатывают в моей сфере, я бы понял сколько действительно стою.

**Знай, с чем работаешь**

Однажды я так стремился работать на одну компанию (мой друг работал именно там и ему все очень даже нравилось), что я забыл задать себе вопрос: “Над чем именно я буду работать?”. Оказалось, что работать я буду в другом отделе, не там, где работает мой друг. Отдел же, не только лишь находился в другом конце здания, но и вообще на другом этаже. В этой суматохе, я даже не успел подумать, действительно ли работа мне подходит.

Еще одна ошибка заключалась в том, что я даже не спросил о среде разработки, инструментах и языках, с которыми я буду работать.

**На работе**

после поступления на работу сложности не заканчиваются. Каждый день будет преподносить новые подлянки, с которыми тебе придется сталкиваться.

**Комментирование кода**

“Мне не нужно комментировать код, читать его легко и ясно” - так я думал, когда только начинал программировать профессионально. Больше я не делаю подобных глупых заявлений.

Комментарии – самая мощная вещь в любом языке программирования. Как только у тебя появляются какие-либо мысли, касающиеся кода, сразу же записывай их в комментариях.

Я прочел бесчисленное количество секций кода, где всего лишь один маленький простой комментарий, способен значительно упростить работу с комплексным кодом и сложными алгоритмами.

Закомментированные строчки кода еще хуже, чем код без комментариев вообще. Когда ты пытаешься разобраться в чужом коде и вдруг натыкаешься на закомментированную секцию кода, работать становится в разы сложнее. Если ты закомментируешь часть кода по какой-либо причине, то при чтении твоего кода другим человеком, ему будет не понятно, с какой целью ты это сделал.

Используйте комментирование разумно. Хорошо расставленные комментарии не только облегчат чтение кода, но и помогут тебе найти ошибки в нем.

Если что-то в коде не совпадает по смыслу с комментарием, то скорее всего, с этим кодом что-то не так.

**Задавайте вопросы**

Если у вас возникла какая-либо проблема и вам нужна помощь, то обратитесь за ней сразу же. Оттягивание просьбы о помощи может привести к неверным решения, или еще хуже, вы можете что-то сломать. Задавайте вопросы сразу же, как они пришли вам в голову, даже если их можно просто загуглить. Опасаться задавать вопрос, потому что вы боитесь выглядеть как полный идиот, не стоит, так как в конечном итоге, вы можете начать писать что-то очень неправильное.

Задавать вопросы – одна из самых важных вещей, который вы можете сделать для ускорения обучения, и для того, чтобы быстрее попасть в какой-нибудь проект. Если ты не будешь задавать вопросы вовремя, то это может привести к поспешным неверным решениям.

**Работа на дому**

Не бойтесь периодически спрашивать о работе на дому. Иногда это отличный способ избавиться от стресса и по-настоящему сфокусироваться на работе. Существуют целый ряд компаний, построенных на принципе работы на дому и это реально работает.

Конечно, также существуют компании, которые работают полностью противоположно. Я работал более года с командой в Австралии, находясь в офисе в Новой Зеландии. Сотрудничество и кооперация проходила в онлайн через email, чат и звонки. Практической разницы от того, где я буду работать (дома или в офисе), не было, но я был вынужден все равно работать в офисе.

Ищите возможность поработать один денек дома или где-нибудь еще, когда это возможно.

**Время, реально потраченное на программирование**

К сожалению, ты не будешь программировать 100% своего времени. Как бы печально тебе от этого не было, знай - в этом нет ничего плохого. На самом деле, программирование это не 100% написание кода.

Большое количество твоего времени, будет потрачено на различные встречи, которые в конечном итоге будут занимать все больше и больше времени. В результате таких встреч, ты будешь писать все меньше и меньше строчек кода, для создания максимально оптимизированного продукта.

**Вне работы**

Чтобы кто не говорил, но правильно проводить свободное время также важно, как и работать

**Программирование в личное время**

Если ты чувствуешь, что застрял в одном типе окружения, знание парочки ведущих технологий, поможет тебе найти выход. Программирование в свободное время даст вам огромный толчок и откроет новые возможности в вашей сфере.

**Чтение**

Сейчас, после прочтения нескольких книг, я пожалел, что не начал читать книги ранее. Существует безграничное количество вещей, которым вы можете научиться благодаря книгам. Просто возьмите парочку книг и начните их читать в перерывах, в офисе или по пути на работу.

**Письмо**

Письмо, может стать отличным способом для продвижения по карьерной лестнице. Хороший блог поможет тебе справиться с проблемами, с которыми ты уже сталкивался до этого. Если ты будешь записывать все сложности, с которыми тебе предстояло столкнуться, то в следующий раз, когда возникнет проблема, ты сможешь обратиться к своим записям за помощью.

В начале это может показаться странным, но записывание различных вещей – отличный способ снять с себя все негодования и скопившийся стресс.

**Занятие спортом**

Первые два года работы, я не выполнял регулярных тренировок, и это, разумеется, повлияло на меня не очень. Как программист, большую часть времени ты будешь сидеть на рабочем месте, смотря в экран. Конечно, как я говорил ранее, тебе не придется программировать весь день, но между встречами и сидением на рабочем месте, двигаться ты будешь все равно не много. Постарайся заниматься спортом так часто, как только ты можешь.

**Бери перерывы**

Насколько важно выделять все время на работу, настолько важно и находить время на отдых. Если по какой-либо причине ты не копишь деньги на большой отпуск, то иногда неплохо делать и без того длинные выходные, еще длиннее, или периодически брать несколько дней для отдыха.

Здесь я совершил очередную ошибку. Пытаясь накопить как можно больше дней для отдыха, вместо того чтобы отдыхать периодически, я полностью вымотался. Работа без выходных положительно влияла на мои финансы, но не на моё физическое и психическое здоровье.

**Спасибо за внимание!**

Я надеюсь, что