

Безопасность и переменны

- 1. Если человек не чувствует, что находится в безопасности, он будет противиться переменам.
- 2. Перемены необходимы руководителю для успешной работы (наверняка они необходимы и в любой другой деятельности).
- 3. Неуверенность заставляет человека избегать риска.
- 4. Избегая риска, человек упускает все новые возможности и выгоды, которые могли бы принести ему переменны.
- 5. Человека легко запугать прямыми угрозами, но также можно просто дать ему понять, что при случае с ним могут обойтись грубо и жестоко. Эффект будет таким же.

Отрицательная мотивация

- 1. Угрозы — самый неподходящий вид мотивации, если вас волнует производительность сотрудников.
- 2. Чем бы вы ни угрожали, задача все равно не будет выполнена, если с самого начала вы отвели на ее выполнение слишком мало времени.
- 3. Более того, если люди не справятся, вам придется выполнить свои обещания.

Части тела, необходимые для управления проектами

- 1. Для руководства нужны сердце, нутро, душа и нюх.
- 2. Так что:
- руководить надо сердцем;
- чувствовать нутром;
- вкладывать в команду и проект душу;
- иметь нюх на всякую ерунду и бессмыслицу.

Главнокомандующий на поле битвы, как метафора управления проектами.

- К началу сражения работа главнокомандующего уже закончена. Собеседование и прием на работу.
- 1. Чтобы нанять человека на работу, менеджеру необходимы все его способности: сердце, душа, нюх и способность чувствовать нутром (по большей части последнее).
- 2. Не пытайтесь нанимать людей в одиночку — гораздо лучше задействовать в этом процессе интуицию двух менеджеров.
- 3. Попросите новых членов команды взяться в проекте за ту работу, которую им уже случалось успешно выполнять в прошлом, а прочие амбиции и рост отложить до следующего проекта.
- 4. Попросите наводку — тот человек, которого вы выбрали себе в команду, наверняка может посоветовать, кого вам еще следует нанять.
- 5. Больше слушайте, меньше говорите.
- 6. И все это сработает еще лучше, если вы слегка подтасуете колоду.

Повышение производительности

- 1. Не существует никаких краткосрочных мер, которые позволили бы быстро повысить производительность работы.
- 2. Повышение производительности — результат долгосрочных усилий.
- 3. Любые средства для повышения производительности, которые обещают немедленный результат, на самом деле не что иное, как «птичье молоко»

Управление рисками

- 1. Чтобы управлять проектом, достаточно управлять его рисками.
- 2. Создайте список рисков для каждого проекта.
- 3. Отслеживайте те риски, которые являются причиной провала проекта, а не только конечные риски.
- 4. Оцените вероятность возникновения и стоимость каждого риска.
- 5. Для каждого риска определите показатель — симптом, по которому можно определить, что риск превращается в проблему.
- 6. Назначьте специального человека для управления рисками, и пусть он не поддерживает девиз «Мы можем все!», который культивирует начальство.
- 7. Создайте доступные (возможно, анонимные) каналы для сообщения плохих новостей руководству.

Играй в защите

- 1. Сокращайте потери.
 - 2. Успех проекта можно скорее обеспечить сокращением ненужных усилий, чем стремлением к новым победам.
 - 3. Чем раньше вы прекратите ненужную работу, тем лучше для всего проекта.
 - 4. Не пытайтесь создавать новые команды без необходимости; поищите в коллективе уже сложившиеся и сработавшиеся команды.
 - 5. Оставляйте команды работать вместе и после окончания проекта (если они сами того хотят), чтобы у пришедших вам на смену руководителей было меньше проблем с плохо срабатывающимися командами.
 - 6. Считайте, что команда, которая хочет продолжать работать вместе и дальше, — это одна из основных целей любого проекта.
 - 7. День, который мы теряем в начале проекта, значит так же много, как и день, потерянный в конце.
8. Есть тысяча и один способ потратить день зря и ни одного, чтобы вернуть этот день обратно.

Моделирование процесса разработки

- 1. Моделируйте свои предположения и догадки о том, как пойдет процесс работы.
- 2. Обсуждайте эти модели вместе с партнером, чтобы лучше понимать процесс работы и вносить необходимые исправления.
- 3. Предсказывайте результаты работы с помощью модели.
- 4. Сравнивайте результаты, полученные в процессе моделирования, с реальными.

Извращенная политика

- 1. В любой момент нужно быть готовым отказаться от работы и попросить расчет...
 - 2. ...однако это не означает, что тем самым вы сумеете избежать последствий извращенной политики.
 - 3. Извращенная политика достанет вас везде, даже в самой здоровой и чистой организации.
4. Главный признак извращенной политики: во главу угла
- ставятся личные цели и влияние, а не общие интересы компании.
5. Это может произойти даже тогда, когда личные цели
- напрямую противоречат целям организации.
6. Один из побочных эффектов извращенной политики: иметь хорошо укомплектованную команду становится небезопасно.

Сбор метрических данных

- 1. Определяйте размер каждого проекта.
- 2. Не усердствуйте поначалу с выбором единицы измерения — если впоследствии вам предстоит работать с реальными данными, для начала сойдут и абстрактные единицы.
- 3. Стройте сложные метрики на основе простых (тех, которые легко подсчитать в любом программном продукте).
- 4. Собирайте архивные данные, чтобы считать производительность труда по уже законченным проектам.
- 5. Работайте над формулами вычисления сложных синтетических метрик до тех пор, пока полученные результаты не будут наиболее точно отражать отношение абстрактных единиц к указанному в архивных данных объему работ.
- 6. Проведите через всю архивную базу данных линию тренда, которая будет показывать ожидаемый объем работ в виде отношения значений сложных синтетических метрик.
- 7. Теперь для каждого нового проекта достаточно будет высчитать значение синтетической метрики и использовать ее при определении ожидаемого объема работ.
- 8. Не забывайте об «уровне помех» на линии производительности и используйте его, как индикатор при определении допустимых отклонений от общей траектории.

Процесс разработки и его улучшение

- 1. Хороший процесс разработки и его постоянное улучшение — весьма достойные цели.
- 2. Но существуют еще и рабочие цели и задачи: хороший работник сконцентрирует внимание как раз на них, даже если вы его об этом не просили.
- 3. Формальные программы, направленные на улучшение существующего процессе разработки, будут дорого стоить команде — и во временном, и в денежном отношении. Даже отдельные усилия по улучшению процесса могут отбросить команду далеко назад. Что касается возможного повышения производительности, то даже если это и произойдет, то едва ли выгоды от этого повышения покроют затраты.

Процесс разработки и его улучшение

- 4. Можно надеяться получить положительный результат от одного какого-нибудь хорошо взвешенного и тщательно выбранного усовершенствования в методике работы. В этом случае оно может покрыть деньги и время, потребовавшиеся на его внедрение.
- 5. Попытка внедрить более одного усовершенствования методологии — гиблое дело. Программы, направленные на улучшение многих приемов и навыков (например, переход на следующий уровень СММ), скорее всего приведут к тому, что сроки только увеличатся.
- 6. Опасность стандартизированного процесса разработки состоит в том, что за рутинными операциями люди могут не заметить возможность сэкономить время и усилия по разработке проекта.
- 7. Что касается чрезмерно больших команд, то там стандартизированный процесс будет неукоснительно соблюдаться до тех пор, пока он позволяет всем чувствовать себя при деле (не важно, с пользой для проекта или нет).

Делать работу по-другому

- 1. Есть только один способ сократить время на разработку, когда его и без того мало — уменьшить сроки отладки программы.
 - 2. Проекты с высокой производительностью требуют гораздо меньше времени на отладку и исправление ошибок.
 - 3. Проекты с высокой производительностью требуют гораздо больше времени на проектирование.
4. Нельзя заставить людей делать что-то по-другому, если ты о них не заботишься, если ты ими не интересуешься. Чтобы они изменились, ты должен понимать (и ценить) их самих, что они делают и к чему стремятся.

Что дает давление сверху

- 1. Люди не начнут быстрее соображать, если руководство будет давить на них.
- 2. Чем больше сверхурочной работы, тем ниже производительность.
- 3. Немного давления и сверхурочной работы могут помочь сконцентрироваться на проблеме, понять и почувствовать ее важность, но длительное давление всегда плохо.
- 4. Возможно, руководство так любит применять давление, потому что просто не знает, как еще можно повлиять на ситуацию, или же потому, что альтернативные решения кажутся им слишком сложными.
- 5. Ужасная догадка: давление и сверхурочная работа призваны решить только одну проблему — сохранить хорошую мину при плохой игре.

Сердитый начальник

- 1. Злость и неуважение заразительны. Когда высшее руководство демонстрирует злость и неуважение к подчиненным, руководители среднего звена начинают копировать их поведение. Точно так же дети, которых наказывали в детстве, часто впоследствии становятся жестокими родителями.
- 2. Неуважение и злоба, по мнению некоторых начальников, должны заставить подчиненных лучше работать. Это типичная политика «кнута и пряника». Но разве когда-нибудь неуважение со стороны начальства приводило к тому, что люди начинали лучше работать?
- 3. Если начальник демонстрирует неуважение к подчиненным, это признак того, что он не может больше занимать свою должность (а вовсе не признак того, что у него плохие подчиненные).

Туманные спецификации

- 1. Неясность спецификации говорит о том, что между участниками проекта есть неразрешенные конфликты.
- 2. Спецификация, в которой нет списка типов входящей и исходящей информации, не должна даже приниматься к рассмотрению. Это значит, что она попросту ничего не специфицирует.
- 3. Никто никогда не скажет вам, что спецификация плоха. Люди скорее будут обвинять себя в неспособности понять написанное, чем ругать авторов спецификации

Конфликт

- 1. Проект, в котором участвуют несколько сторон, обязательно столкнется с конфликтом интересов.
- 2. Процесс создания и распространения программных систем — прямо-таки рассадник всевозможных конфликтов.
- 3. В большинстве компаний, где создается программное обеспечение, никто специально не занимается вопросом решения конфликтов.
- 4. Конфликт заслуживает понимания и уважительного отношения. Конфликт не имеет ничего общего с непрофессиональным поведением.

Конфликт

- 5. Донесите до каждого, что постараетесь учитывать интересы всех участников, и проследите, чтобы так оно и было.
- 6. Тяжело договариваться. Гораздо легче выступать посредником.
- 7. Объявите заранее, что если интересы конфликтующих сторон полностью или частично противоположны, то поиск решения будет переложен на посредника.
- 8. Не забывайте: мы находимся по одну сторону баррикад. По другую сторону находится сама проблема

Кто такой катализатор проекта

- 1. Существуют люди-катализаторы. Они помогают создать здоровую команду, отношения, боевой дух. Даже если бы они больше ничего не делали (а как правило, они делают куда как много), их роль в проекте остается одной из наиболее важных.
- 2. Посредничество — еще одна сфера, в которой люди-катализаторы просто незаменимы. Впрочем, посредничеству можно научиться, это не очень сложно.
- 3. Первым шагом к посредничеству должна быть маленькая церемония. Например, можно произнести фразу: «Вы позволите мне попробовать помочь вам решить этот спор?»

Человеку свойственно ошибаться

- 1. Нам кажется, что самое страшное — не знать чего-то. На самом деле гораздо хуже быть уверенным, что знаешь, когда на самом деле это не так.

О персонале

- 1. Если в самом начале проект делает большая команда, это снижает эффективность самой ответственной части работы — определения архитектуры системы (потому что всем разработчиком надо скорее дать какую-то работу).
- 2. Если работу раздать людям и командам еще до того, как завершится стадия дизайна продукта, не получится создать простые и эффективные модели взаимодействия между людьми и рабочими группами.
- 3. Это приведет к потере независимости, увеличению числа собраний и совещаний, общему недовольству.
- 4. В идеале было бы хорошо сначала набрать маленькую команду, которая создала бы продуманную архитектуру системы, а уже потом, на последнюю, шестую часть времени разработки в эту команду можно было бы добавить новый персонал (который работал бы непосредственно над кодированием).
- 5. Ужасное предположение: кажется, те команды, перед которыми не ставят жестких сроков, заканчивают работу быстрее!

Проблемы социологии

- 1. Собрания должны быть небольшими. Для этого нужно сделать так, чтобы люди не боялись пропускать ненужные им собрания. Самый простой способ — заранее опубликовать повестку дня, а потом всегда строго ее придерживаться.
- 2. Каждому проекту нужна какая-то церемония или ритуал.
- 3. С помощью церемоний можно концентрировать внимание собравшихся на основных целях и задачах совещания: сократить состав рабочей группы, повысить качество программного кода и т. п.

Проблемы социологии

- 4. Защищайте людей от оскорблений и ругани Начальства.
- 5. Запомните: в работе страх = гнев. Те руководители, которые любят кричать на своих подчиненных и всячески унижают и оскорбляют их, на самом деле просто чего-то очень боятся.
- 6. Наблюдение: если бы для всех проявление грубости и злостик подчиненным всегда значило бы, что начальник просто боится, то

никто из начальников не стал бы так себя вести просто из страха, что его страх станет заметен! (Это, конечно, не решает проблем такого руководителя, но, по крайней мере, оберегает его подчиненных.)

О патологической политике (еще раз)

- 1. Эту патологию невозможно вылечить снизу.
- 2. Не стоит терять время или подвергать себя опасности, чтобы проверить предыдущий постулат на собственном опыте.
- 3. Иногда единственным выходом из ситуации становится выжидание. Попробуйте подождать, пока проблема не разрешится сама по себе или пока вы не найдете способ уйти от нее в сторону.
- 4. Чудеса, конечно, случаются, но лучше все же на них не рассчитывать.

Злоба и скупость

- 1. Злоба и скупость — вот формула, которую начинают применять в плохих компаниях те, кто несет ответственность за неудачи в бизнесе.
- 2. Злоба и скупость прямо противоположны истинным целям любой хорошей компании — быть щедрыми и заботливыми по отношению к своим сотрудникам.
- 3. Когда вы подмечаете в компании проявления злобы и скупости, знайте, их настоящая причина — страх и боязнь провала.

Основы здравого смысла

- 1. У проекта должно быть два срока сдачи — запланированный и желаемый.
- 2. Эти сроки должны быть разными