

Марш победителей. Семь принципов проектного управления

Азбука с картинками для начинающих менеджеров проектов

Знание нескольких принципов освобождает от запоминания множества фактов.

Клод Гельвеций



Здесь я написал о главном принципе управления программным проектом. О «Принципе 4П». Не люди должны строиться под выбранную модель процесса, а модель процесса должна гибко подстраиваться под конкретную команду, проект и продукт, чтобы обеспечить наивысшую эффективность.

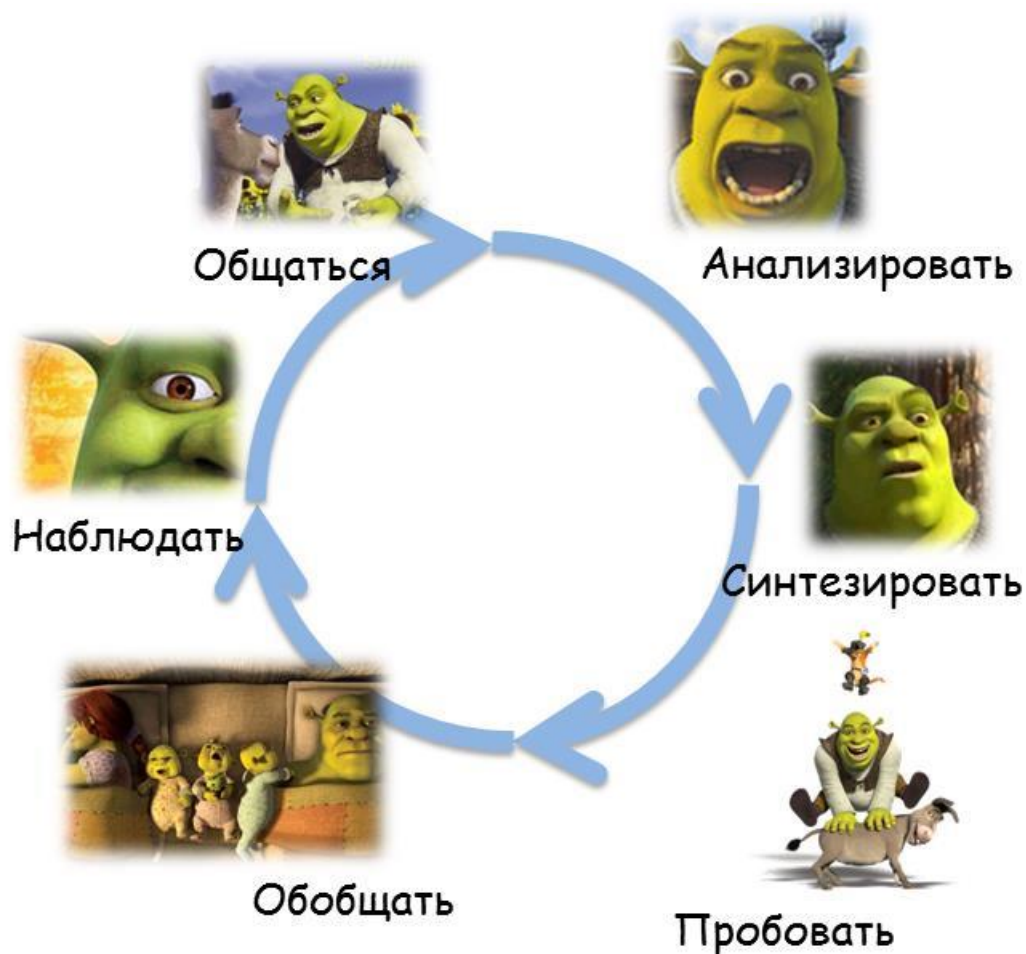
Это не единственный принцип эффективного управления проектом разработки ПО. Есть и другие.

Их всего семь. Итак...

Принцип 2. Принцип достаточного разнообразия

Для «хорошего» управления количество возможных состояний управляющего устройства (разнообразие) должно быть не меньше, чем количество состояний объекта управления. (С) У.Р.Эшби “Введение в кибернетику” М., ИЛ, 1959

Работа менеджера – ежедневный цикл: наблюдать, общаться, анализировать и выявлять первопричину проблем, синтезировать корректирующее воздействие, пробовать и обобщать полученный результат.



Сколько людей и ситуаций, столько и вариантов решений должен иметь эффективный руководитель в своем арсенале. «Если у руководителя в руках только молоток, то все вокруг будут похожи на гвозди». Понять человека можно, только слушая и слыша, что он говорит. Руководитель, который в течение недели не пообщался индивидуально с каждым из своих прямых подчиненных, зря получает зарплату.

Принцип 3. Четыре условия эффективной работы



Для того чтобы сотрудник мог эффективно решить поставленную руководителем задачу, необходимо и достаточно, чтобы он:

1. Понимал цель работы.
2. Умел ее делать.
3. Имел возможность ее сделать.
4. Хотел ее сделать.

Ни одна задача не будет решена за любое, отведенное на это время, если человек не захочет ее сделать. У каждого участника рабочей группы должна быть личная цель (внутренняя мотивация), которую он сможет достичь, продвигая проект к успеху. Если у сотрудника нет такой личной цели, не стоит его брать в проект.

Принцип 4. Четыре роли руководителя

Для того чтобы обеспечить четыре условия эффективной работы руководитель должен уметь исполнять четыре роли:



Штурман-направляет



Наставник-обучает



Помощник-обеспечивает



Вдохновитель-мотивирует

Помним, наставник это не тот, кто учит, а тот, у кого учатся. Добиться от участника приверженности проекту больше, чем имеется у руководителя, никому не удавалось.

Принцип 5. Принцип лидерства

Руководитель программного проекта должен стать лидером, вокруг которого сплотится эффективная команда.



В разработке ПО многие уже признали, что наиболее эффективные процессы взаимодействия складываются в самоуправляемых и самоорганизующихся рабочих командах. Но, как не бывает лидеров без последователей, так и не бывает команд без лидеров. Лидера нельзя назначить. Лидер должен получить от команды признание своих профессиональных качеств и завоевать полное доверие.

Принцип 5. Четыре стратегии лидерства

В зависимости от готовности каждого участника рабочей группы выполнять задания руководителя, он должен использовать одну из 4-х стратегий:

Профессиональное признание

Да



Участие



Делегирование

Нет



Директивное управление

Нет



Объяснения

Да

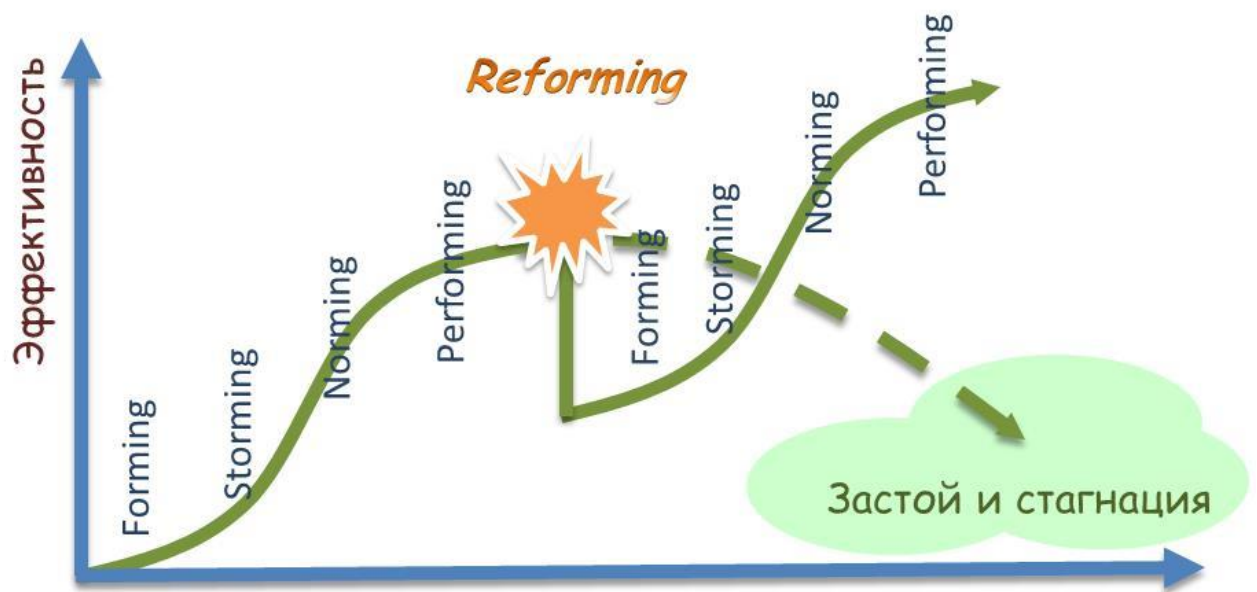
Доверие команды

Не существует одной лучшей стратегии руководства, стиль лидерства должен определяться конкретной ситуацией. Чтобы получить признание профессиональной компетенции лидер должен сочетать директивное управление и объяснения - "продавать" свои решения, убеждать в их правильности. Завоевать доверие можно, если поощрять коллективное принятие решений, обмен идеями, поддерживать инициативу сотрудников. Делегирование возможно только при взаимном доверии.

Принцип 7. Принцип цикличности

Четыре фазы развития команды: формирование, конфликты, становление, отдача, - должны циклически повторяться, чтобы обеспечивать непрерывный рост эффективности.

Если руководитель не будет прилагать дополнительные усилия команда, рано или поздно, начнет «сползать» с плато наивысшей эффективности в состояние застоя и стагнации.



Руководитель должен постоянно искать ответы на вопросы: «Что лишнее мы делаем?»; «Что можно делать проще?»; «Что угрожает проекту?», - работать на сокращение ненужных усилий вместо того, чтобы «стремиться к новым героическим подвигам». Необходимо перестраиваться, отказываться от того, что перестало действовать или стало работать неэффективно. Разумеется, делать это следует, после сдачи очередного релиза программного продукта, ну и, возможно, в случае глубокого кризиса проекта.

Заключение

Одна из книг Эдварда Йордана называется "Смертельный марш" (Полное руководство для разработчика программного обеспечения по выживанию в безнадежных проектах).

Уверен, можно жить не по Йордану. Хорошо управляемый проект может быть успешно выполнен обычной командой разработчиков.