

## Лекция от 03.03.2016

### Разработка плана управления проекта

Содержание презентации:

Что такое план управления проектом

Когда его разрабатывать

Кто его разрабатывает

Состав плана

План управления проектом является не только план графиком.

Он описывает все аспекты проекта.

Может быть разной степени формализации – от полного формального документа до устных обсуждений. В начале проще работать с электронным документом.

Есть много возможных методологий в управлении проектом, это должно входить в план управления проектом.

Первоначальный вариант создается на ранних этапах, либо когда готов старт проекта.

Постоянно происходит изменения и уточнения плана, он должен поддерживаться в актуальном состоянии.

В идеальном случае, когда готов устав, должен быть руководитель, но это бывает не всегда.

Он должен организовать все работы по созданию плана управления проектом.

Отдельные разделы могут создаваться высшим руководством, командой, заказчиком.

В ряде проектов план должен быть согласован с заказчиком (просмотр, правки, тестирование чувствительных мест).

Менеджер отвечает за результат и согласованность.

//документ будет распространяться без автора, поэтому нужно хорошее название (должно быть указано в тексте, в имени файла), номера страниц, авторские данные и положение в тексте на каждой странице, содержание.

Состав:

- О документе (желательная метаинформация)
  - Цель и рамки документа
  - Термины и определение
  - Ссылки на другие документы (на внешние источники и на распределенные части ПУП)
- Определение проекта
  - Идентификация проекта (название, заказчик, исполнитель)
  - Постановка задачи
  - Цели проекта (явные и скрытые) – позволяет более точно понять, что такой успех или неуспех проекта. В сложных случаях коммерческий провал не означает полный провал проекта, может быть получен результат по одной из скрытых целей.
  - Рамки проекта – что надо и не надо делать в этом проекте в общих чертах.
  - WBS (структурная декомпозиция проекта) – деление крупных задач на более мелкие в виде дерева, уточняет рамки проекта.
  - Сроки проекта
  - Основные фазы – важно для больших проектов.
  - Основные ограничения (стоимость, ресурсы) – может быть ограниченный доступ к уникальным ресурсам.

- Критерии и процедуры приемки (чем раньше их поставить, тем лучше, прием могут проводить не те люди, с которыми были взаимодействия в процессе разработки, может быть долгий процесс или без присутствия исполнителя)
- **Организационная структура проекта**
  - Заинтересованные люди и их ответственности  
Состав – заказчик, исполнитель, директор, пользователь, подразделения фирмы, скрытые. МБ активно поддерживающие, позитивные, нейтральные, заинтересованные в провале, способствующие провалу. Надо знать, кто, как и на что влияет в проекте.  
Ответственности – принятие стратегических решений, предоставление ресурсов, утверждение планов, выработка и утверждение технических решений.  
Можно описать, что можно сделать, чтобы люди становились более лояльны к проекту.
  - Команда проекта - можно не по именам, а по должностям, если не все работники еще наняты.
  - Структура команды – группы разработки, тестирование и т. д.
  - Роли участников и обязанности – руководитель, архитектор, программист, тестировщик и т. д. Для них надо описать, что от них ожидается.
  - Поименный список участников.
- **Ресурсы проекта**
  - Оборудование – зависит от задачи, может быть редкое и нестандартное, требующие дополнительных ресурсов и условий эксплуатации.
  - ПО – цена ПО для разработки может не покрыться стоимостью заказа.
  - Финансовые ресурсы - командировки, субподряды.
- **Управление конфигурацией**  
Конфигурационное управление – постоянная деятельность, направленная на обеспечение целостности всех рабочих продуктов проекта.  
В плане управления проектом представлена планом управлением конфигурацией.
- **Основные процедуры управление проектом**
  - Перечень типов отчетов (с шаблонами и временными метками)
  - Методы планирования – какими средствами пользоваться при планировании
  - Процедуры управления изменениями  
Самое частое изменение – это изменение требований. Заказчик всегда заинтересован сделать больше, надо соблюдать баланс.  
Есть рамки проекта, круг лиц, принимающих решения и т. д.
  - Процедуры анализа хода проекта  
Нужны какие-нибудь методы, позволяющие понять, насколько эффективно выполняется проект. Должны показывать настоящее положение дел.
- **План управления коммуникациями**
  - участники
  - способы
  - график коммуникации
  - технические средства коммуникации
- **План разработки**

#### **Фазы проекта и их результаты**

//Пример стандартного поверхностного разбиения

- Старт – сформирована команда, подготовлены рабочие места.
- Разработка – основной функционал, документы.
- Стабилизация – кандидатская версия.
- Релиз – готовый пакет.

#### **План-график(диаграмма Ганта):**

Этапы

Ресурсы

## Зависимости задач

### Тестирование в проекте

- Концепции тестирования - виды тестирования, ответственный, статусы ошибок
- Документы тестирования - план тестирования, тесты, отчеты
- Взаимоотношения тестеров и разработчиков - независимое тестирование, как часть команды, кто принимает решение о статусе ошибок

### Управление рисками

- План управления рисками – основные источники опасности, периодичность мониторинга, вероятности, основные виды реакции.
- Выявленные риски – 10 топ рисков, план реагирования, триггеры.

### Передача результатов проекта

- График передачи результатов – точки завершения этапов, по календарным месяцам, по завершению проекта и т.д.
- Правила передачи результатов
  - Что входит в пакет - инсталляция, исходные коды, сопроводительное письмо, список открытых/закрытых ошибок, реализованная функциональность, результаты тестирования, отчеты тестирования.
  - Способ передачи – интернет, диск/почта, из исходных кодов

Лекция от 10.03.2016

## Предварительная работа по проекту

### Содержание

- Запрос от заказчика
- Подготовка предложения
- Структурная декомпозиция работ
- Оценка проекта
- Календарный план-график
- Согласование условий
- Заключение договора

### Виды запросов:

RFI (Request For Information) – анкета с общими вопросами:

- Прибыль за последние несколько лет – нет проблем для публичных компаний, может быть высчитывается через косвенные показатели, может быть подписан договор о неразглашении.
- Собственники
- Основные рынки
- Количество сотрудников
- Наличие профессиональных сертификатов
- Конкретные ссылки на успешные проекты и довольных заказчиков
- Условия проверки качества проекта

RFP (Request For Proposal) – описание потребностей заказчика в конкретном

проекте/услуге/продукте. Запрос может быть гипотетический, на проект, для которого не нужна реализация, для проверки реакции исполнителя.

### Предметы запросов

Разработка решения/продукта:

- ПО коммутатора GSM
- Сайт интернет-банка
- Игра для смартфонов

Оказание услуг:

- Сопровождение ПО
- Создание центра разработки(ODC)

**Предложения по разработке решения** – формальный документ. Все, что в нем описано, надо соблюдать. Часто выставляют срок действия предложения (в большинстве случаев около месяца).

Рамки задачи

- Краткое описание проекта – частично повторяет требования заказчика.
- Требования

Описание решения

- Архитектура
- Спецификация среды разработки – доказательство профессионализма исполнителя.
- Основные вехи и результаты – предоставление материальных доказательств продвижения работ, защита заказчика от зависания работы над продуктом.

Процедуры – детальность уточняется в зависимости от многих факторов.

- Политики взаимодействий/коммуникаций/отчетов
- Описание поставок и приемок
- Описание тестового решения

Оценка – обязывающий прогноз, надо рассчитывать очень внимательно.

- Оценка объема работ
- Оценка сроков работ
- Оценка команды
- Оценка стоимости

Условия оплаты – по факту затраченных ресурсов или фиксированная цена, от этих условий может зависеть содержание остального предложения. В первом случае оценка и отслеживание более детальны и точны.

### **Рамки проекта**

- Описание продукта – краткое.
- Список продуктов, компонентов, и т. д.
- Цели проекта - **измеримые** критерии, достигнув которые можно считать, что проект завершен успешно. Как минимум стоимость, качество и время.

### **Структурная декомпозиция работ (WSB)**

*Для чего нужна:*

Детально описать рамки проекта.

Используется как критерий: чего там нет, того нет и в проекте.

Используется для оценки проекта.

Для составления календарного плана проекта.

*Основные критерии:*

Представлено все, что нужно делать.

Делит проект на меньшие управляемые элементы.

Содержит более детальное определение работ на более низких уровнях.

Сама работа находится на самых нижних элементах структуры, иногда называется рабочими пакетами.

Работу в листьях легко оценить.

*Как создавать:*

1. Выбрать главные элементы поставки, обычно отражают как проект будет организован.
2. Решить, можно ли оценить стоимость и длительность работ. Если да, то перейти в пункт 4, иначе в пункт 3.
3. Определить основные подкомпоненты, они должны быть определены в виде достижимого и верифицируемого результата.
4. Проверить корректность оценок:
  - Все ли подкомпоненты необходимы и достаточны для выполнения работы? Если нет, то изменить соответственно требованиям.
  - Каждый элемент ясно и точно определен? Если нет – описать
  - Можно ли спланировать, оценить бюджет, назначить ответственного за выполнения работы? Если нет, провести анализ.

## **Оценка проекта**

*Что оценивать?*

Объем – сколько единиц работы необходимо сделать, чтобы выполнить проект.

Трудоемкость – сколько труда надо вложить.

Сроки – сколько календарного времени надо на завершение проекта.

Стоимость – какое вознаграждение требуется для выполнения работы.

*В чем измерить объем?*

- Идеальные человеко-дни.
- Строки кода.
- Экраны – количество и качество пользовательских экранов.
- Точки перехода API – сколько различных вызовов могут получать приложения.
- Use Case Point – консолидированный критерий из всех предыдущих.

## **Лекция от 17.03.2016**

### **Сложности оценки**

Нет опыта оценки

Не до конца понятно, что делать

Большой объем работы, сложно подступиться

Непонятно, кто будет выполнять работу

Не хватает знаний предметной области

Не хватает знаний технологий

Детали могут сильно менять сложность

Нефункциональные требования – более глобальные качества продукта такие как кроссплатформенность, большое количество пользователей.

### **Конус неопределенности оценки**

Сложно заранее оценить проект, оценка будет приблизительная в рамках конуса неопределенности.

Initial Project Concept – 0.25x

Approved Product definition – 0.5x

Requirement Complete – 0.6x

User Interface Design Complete – 0.7x

Detailed Design Complete – 0.8x

Software Complete – 1x

### Приемлемая точность оценки

- Предварительная оценка +/-50%  
Начальный этап переговоров, точность оценки декларируется
- Стартовая оценка +/-25%  
Основа для предложения, точность оценки оценивается и закладывается в буфер при планировании
- Уточненная оценка +/-15%  
После формирования образа системы, может быть явно предусмотрена договором

### Методы оценки проекта

- Оценка по аналогу – оценка по собственному опыту или опыту людей, выполнявших подобную работу. Подходит для небольших однотипных проектов.
- Параметризация – декомпозиция и оценка по частям.
- Planning poker – для планирования коротких итераций, участвуют все участники проекта, хорошо прорабатываются требования, все участники получают карточки с возможными оценками задачи. Модератор выносит задачи, которые надо сделать команде, каждый член которой оценивает задачу с помощью карточек. Если оценки очень разные, то задача обсуждается и уточняется, пока оценки не станут близкими. Далее по формуле из суммы оценок задачи высчитывается некоторая более реальная оценка (например, в трудоемкости).
- PERT:  $(O+4R+P)/6$  – делать 3 оценки: оптимистичную, пессимистичную, реалистичную. Потом усреднить их, подставляя в формулу.
- Wideband Delphi – уточнение и консолидация мнений экспертов об оценке. Каждый эксперт выдает оценку, далее идет обсуждение оценок, пока не сойдутся в примерно одинаковой оценке, но жестко ограничиваем количество итераций до одной.
- Экспертная оценка  
Эта оценка хорошо работает в небольших и средних проектах.
  1. Функциональная декомпозиция системы – случаи использования, мокапы интерфейса.
  2. Анализ внешней среды системы – список допущений, связанный с внешней средой и пользователями.
  3. Выявление нефункциональных требований
  4. Оценка полноты и непротиворечивости требований
  5. Высокоуровневая структурная декомпозиция – разбор с точки зрения архитектуры.
  6. Выбор технологии и инструментария – зависит от требований.
  7. Проверка основных решений на прототипах и уточнение архитектуры – в первую очередь для наиболее рискованных задач, чтобы было время исправить. После этого этапа есть четкое понимание, что и как надо делать.
  8. Оценка конкретных задач  
Такая оценка утончается с помощью привлечения большего количества экспертов, сравнения с оценками, полученными другими путем.
- Use Case Points (UCP)
  - Актеры и взаимодействия (UAW)  
Открытый API – 1балл  
Протокол – 2 балла  
GUI – 3 балла

- Случаи использования (UUCW)
  - 1-3 шага – 5 баллов
  - 4-7 шагов – 10 баллов
  - 8+ шагов – 15 баллов
- Фактор технической сложности (TCF)
  - 13 факторов с весом от 0,5 до 2
  - Каждому оценка важности от 0 до 5
  - $TCF = 0,6 + \text{суммарный вес} / 100$
- Фактор сложности окружения (ECF)
  - 8 факторов от -1 до 2
  - Каждому оценка важности от 0 до 5
  - $ECF = 1,4 + (-0,3 * \text{суммарный вес})$

$$UCP = (UAW + UUCW) * TCF * ECF$$

Трудозатраты =  $UCP * X$  человеко-часов (X примерно 20-30, подбирается из опыта работы с командой)

*Купить или сделать?*

Для каждого элемента WBS нужно сделать выбор – покупать или создавать?

Критерии – стоимость, время, наличие ресурсов.

#### **Типовые пропуски в оценке**

- Разработка требований
- Разработка системы развертывания и собственно развертывание
- Создание пользовательской и технической документации и справки
- Внедрение в процессы заказчика
- Миграция данных
- Интеграция со сторонними системами
- Обновление «на лету» с сохранением данных
- Интеграционное и системное тестирование
- Передача кода во владение заказчика