

Глава 7: Управление программной инженерией

Из SWEBOK

Содержание

- 1 Инициация и определение области действия
 - 1.1 Определение и обсуждение требований
 - 1.2 Анализ осуществимости
 - 1.3 Процесс обзора и пересмотра требований
- 2 Планирование программного проекта
 - 2.1 Планирование процесса
 - 2.2 Определение результатов
 - 2.3 Усилия, график и оценка стоимости
 - 2.4 Распределение ресурсов
 - 2.5 Управление рисками
 - 2.6 Управление качеством
 - 2.7 Управление планом
- 3 Принятие решения о программном проекте
 - 3.1 Реализация планов
 - 3.2 Приобретение программного обеспечения и управление контрактами с поставщиками
 - 3.3 Реализация процесса измерения
 - 3.4 Мониторинг процесса
 - 3.5 Процесс управления
 - 3.6 Ответственность
- 4 Обзор и оценка
 - 4.1 Определение удовлетворения требований
 - 4.2 Обзор и оценка производительности
- 5 Закрытие
 - 5.1 Определение закрытия
 - 5.2 Действия по закрытию
- 6 Измерение программной инженерии
 - 6.1 Установление и поддержание обязательств по измерению
 - 6.2 Планирование процесса измерения
 - 6.3 Выполнение процесса измерения
 - 6.4 Оценка измерения
- 7 инструментов управления программной инженерией

АКРОНИМЫ

Руководство	<i>Руководство по своду знаний по управлению проектами</i>
PMBOK®	
SDLC	Жизненный цикл разработки программного обеспечения
СЭМ	Управление программной инженерией
SQA	Гарантия качества программного обеспечения
SWX	<i>Расширение программного обеспечения для руководства PMBOK®</i>
WBS	Структура работы

ВВЕДЕНИЕ

Управление программной инженерией можно определить как применение управленческой деятельности — планирование, координация, измерение, мониторинг, контроль и отчетность¹ — для обеспечения того, чтобы программные продукты и услуги программной инженерии предоставлялись эффективно, результативно и на благо заинтересованных сторон. Связанная с ней дисциплина управления является важным элементом всех областей знаний (ЗЗ), но она, конечно, более актуальна для данной ОП, чем для других ОП. Измерение также является важным аспектом всех КА; тема измерительных программ представлена в этом КА.

В каком-то смысле должна быть возможность управлять проектом разработки программного обеспечения так же, как управляются другие сложные проекты. Однако существуют аспекты, специфичные для программных проектов и процессов жизненного цикла программного обеспечения, которые усложняют эффективное управление, включая следующие:

- Клиенты часто не знают, что нужно или что осуществимо.
- Клиенты часто недооценивают сложности, присущие разработке программного обеспечения, особенно в отношении влияния меняющихся требований.
- Вполне вероятно, что более глубокое понимание и изменяющиеся условия приведут к появлению новых или измененных требований к программному обеспечению.
- В результате изменения требований программное обеспечение часто создается с использованием итеративного процесса, а не в виде последовательности закрытых задач.
- Разработка программного обеспечения обязательно включает в себя творчество и дисциплину. Поддержание надлежащего баланса между ними иногда бывает затруднено.
- Степень новизны и сложности часто высока.
- Зачастую основная технология меняется очень быстро.

Действия по управлению программной инженерией происходят на трех уровнях: управление организацией и инфраструктурой, управление проектами и управление программой измерений. Последние два подробно описаны в этом описании КА. Однако это не умаляет важности организационных и инфраструктурных вопросов управления. Общеизвестно, что менеджеры по организационной инженерии программного обеспечения должны быть знакомы с управлением проектами и знаниями в области измерения программного обеспечения, описанными в этом КА. Они также должны обладать некоторыми знаниями в целевой области. Точно так же было бы полезно, если бы менеджеры сложных проектов и программ, в которых программное обеспечение является компонентом системной архитектуры, знали о различиях, которые программные процессы вносят в управление проектами и оценку проекта.

Другие аспекты организационного управления оказывают влияние на разработку программного обеспечения (например, организационная политика и процедуры, которые обеспечивают структуру, в которой выполняются проекты разработки программного обеспечения). Эти политики и процедуры могут нуждаться в корректировке в соответствии с требованиями эффективной разработки и сопровождения программного обеспечения. Кроме того, может потребоваться наличие или установление ряда политик, специфичных для разработки программного обеспечения, для эффективного управления разработкой программного обеспечения на организационном уровне. Например, политики обычно необходимы для установления конкретных общеорганизационных процессов или процедур для задач разработки программного обеспечения, таких как разработка программного обеспечения, создание программного обеспечения, оценка, мониторинг и отчетность.

Другим важным аспектом организационного управления является политика и процедуры управления персоналом для найма, обучения и наставничества персонала для развития карьеры не только на уровне проекта, но и для долгосрочного успеха организации. Персонал, занимающийся разработкой программного обеспечения, может создавать уникальные проблемы с обучением или управлением персоналом (например, поддержание актуальности в условиях, когда базовая технология претерпевает быстрые и непрерывные изменения).

Управление коммуникациями также часто упоминается как упускаемый из виду, но важный аспект работы отдельных лиц в области, где необходимо точное понимание потребностей пользователей, требований к программному обеспечению и конструкции программного обеспечения. Кроме того, желательно управление портфелем, которое обеспечивает общее представление не только о программном обеспечении, разрабатываемом в настоящее время в различных проектах и программах (интегрированные проекты), но также о программном обеспечении, запланированном и используемом в настоящее время в организации. Кроме того, повторное использование программного обеспечения является ключевым фактором в поддержании и повышении производительности и конкурентоспособности. Эффективное повторное использование требует стратегического видения, отражающего преимущества и недостатки повторного использования.

В дополнение к пониманию аспектов управления, на которые однозначно влияют проекты по программному обеспечению, инженеры-программисты должны иметь некоторые знания о более общих аспектах управления, которые обсуждаются в этом КА (даже в первые несколько лет после выпуска).

Атрибуты организационной культуры и поведения, а также управление другими функциональными областями предприятия оказывают влияние, хотя и косвенно, на процессы разработки программного обеспечения в организации.

Обширную информацию об управлении программными проектами можно найти в Руководстве по своду знаний по управлению проектами (Руководство PMBOK®) и Программном расширении к Руководству PMBOK® (SWX) [1] [2]. Каждое из этих руководств включает десять КА по управлению проектами: управление интеграцией проекта, управление содержанием проекта, управление временем проекта, управление стоимостью проекта, управление качеством проекта, управление человеческими ресурсами проекта, управление коммуникациями проекта, управление рисками проекта, управление закупками проекта и заинтересованные стороны проекта. Каждый КА имеет прямое отношение к этому КА управления программной инженерией.

Дополнительная информация также содержится в других ссылках и дополнительной литературе для этого КА.

Этот КА управления программной инженерией состоит из процессов управления программным проектом в первых пяти темах на рисунке 7.1 (инициация и определение содержания, планирование программного проекта, введение в действие программного проекта, обзор и оценка, закрытие), а также измерение программной инженерии в шестой теме и Средства управления программной инженерией в седьмой теме. Хотя управление проектами и управление измерениями часто рассматриваются как отдельные понятия, и действительно каждое из них обладает многими уникальными свойствами, тесная взаимосвязь привела к комбинированному рассмотрению в этом КА.

К сожалению, в индустрии программного обеспечения распространено мнение, что программные продукты поставляются с опозданием, с превышением бюджета, низкого качества и с неполной функциональностью. Управление на основе измерений — основной принцип любой настоящей инженерной дисциплины (см. «Измерение в инженерных основах КА») — может помочь улучшить

восприятие и реальность. По сути, управление без измерения (качественного и количественного) предполагает отсутствие дисциплины, а измерение без управления предполагает отсутствие цели или контекста. Эффективное управление требует сочетания измерений и опыта.

Здесь приняты следующие рабочие определения:

- *Менеджмент* – это система процессов и средств контроля, необходимых для достижения стратегических целей, поставленных организацией.
- *Измерение* относится к присвоению значений и меток рабочим продуктам, процессам и ресурсам разработки программного обеспечения, а также моделям, которые выводятся из них, независимо от того, разработаны ли эти модели с использованием статистических данных.

или другие методы [3*, с7, с8].

Разделы управления проектами разработки программного обеспечения в этом КА широко используют раздел измерения разработки программного обеспечения.

Этот КА тесно связан с другими в *Руководстве SWEBOOK*, и чтение следующих описаний КА в сочетании с этим будет особенно полезным:

- Инженерные основы КА описывает некоторые общие концепции измерения, которые непосредственно применимы к программной инженерии.

Измерительный участок этого КА. Кроме того, концепции и методы, представленные в разделе «Статистический анализ» книги «Основы инженерного дела», напрямую применимы ко многим темам этого КА.

- Требования к программному обеспечению КА описывают некоторые действия, которые должны быть выполнены на этапе инициации и определения содержания проекта.
- КА управления конфигурацией программного обеспечения занимается идентификацией, контролем, учетом состояния и аудитом конфигураций программного обеспечения.

а также средства управления выпуском и доставкой программного обеспечения и управления конфигурацией программного обеспечения.

- Процесс разработки программного обеспечения КА описывает модели жизненного цикла программного обеспечения и отношения между процессами и рабочими продуктами.
- КА качества программного обеспечения подчеркивает качество как цель управления и как цель многих видов деятельности по разработке программного обеспечения.
- КА по экономике программной инженерии обсуждает, как принимать решения, связанные с программным обеспечением, в бизнес-контексте.

РАЗБИВКА ТЕМ ПО УПРАВЛЕНИЮ ПРОГРАММНОЙ ИНЖЕНЕРИЕЙ

Поскольку большинство моделей жизненного цикла разработки программного обеспечения требуют схожих действий, которые могут выполняться по-разному, разбивка тем основана на действиях. Эта разбивка показана на рис. 7.1. Элементы разбивки верхнего уровня, показанные на этом рисунке, представляют собой действия, которые обычно выполняются при управлении проектом разработки программного обеспечения, независимо от модели жизненного цикла разработки программного обеспечения (см. Модели жизненного цикла программного обеспечения в процессе разработки программного обеспечения КА), которые был выбран для конкретного проекта. В этой разбивке нет цели рекомендовать конкретную модель жизненного цикла. Разбивка подразумевает только то, что происходит, и не подразумевает, когда, как или сколько раз происходит каждое действие. Семь тем:

- Инициация и определение масштаба, которые связаны с решением приступить к проекту разработки программного обеспечения;
- Планирование проекта программного обеспечения, в котором рассматриваются действия, предпринимаемые для подготовки к успешному проекту разработки программного обеспечения с точки зрения управления;
- Регламент разработки программного обеспечения, который касается общепринятых действий по управлению разработкой программного обеспечения, которые происходят во время выполнения

- проекта разработки программного обеспечения;
- Обзор и оценка, которые касаются обеспечения того, чтобы технические, календарные, стоимостные и качественные инженерные работы были удовлетворительными;
- Закрытие, в котором рассматриваются действия, выполненные для завершения проекта;
- Измерение в области разработки программного обеспечения, которое занимается эффективной разработкой и внедрением программ измерения в организациях, занимающихся разработкой программного обеспечения;
- Инструменты управления разработкой программного обеспечения, в котором описывается выбор и использование инструментов для управления проектом разработки программного обеспечения.

1 Инициация и определение области действия

Основное внимание в этих мероприятиях уделяется эффективному определению требований к программному обеспечению с использованием различных методов сбора информации и оценке осуществимости проекта с различных точек зрения. После того, как осуществимость проекта установлена, оставшиеся задачи в рамках этого раздела включают спецификацию требований и выбор процессов для пересмотра и обзора требований.

1.1 Определение и обсуждение требований

[3 , с3]

Определение требований и согласование требований устанавливают видимые границы набора выполняемых задач (см. Требования к программному обеспечению КА). Действия включают выявление требований, анализ, спецификацию и проверку. Следует выбирать и применять методы и приемы с учетом различных точек зрения заинтересованных сторон. Это приводит к определению масштаба проекта для достижения целей и удовлетворения ограничений.

1.2 Анализ осуществимости

[4 , с4]

Целью технико-экономического обоснования является разработка четкого описания целей проекта и оценка альтернативных подходов, чтобы определить, является ли предлагаемый проект наилучшей альтернативой с учетом ограничений технологий, ресурсов, финансов и социальных/политических соображений. Следует подготовить первоначальное описание проекта и содержания продукта, результаты проекта, ограничения продолжительности проекта и оценку необходимых ресурсов.



Рисунок 7.1: Разбивка тем для управления программной инженерией КА

Ресурсы включают в себя достаточное количество людей, обладающих необходимыми навыками, помещениями, инфраструктурой и поддержкой (внутренней или внешней). Анализ осуществимости часто требует приблизительной оценки усилий и затрат на основе соответствующих методов (см. раздел 2.3 «Усилия, график и оценка затрат»).

1.3 Процесс обзора и пересмотра требований

[3 , с3]

Учитывая неизбежность изменений, заинтересованные стороны должны договориться о средствах, с помощью которых требования и объем должны быть рассмотрены и пересмотрены (например, процедуры управления изменениями, ретроспективы итеративного цикла). Это ясно подразумевает, что объем и

требования не будут «высечены в камне», но могут и должны пересматриваться в заранее определенные моменты по мере развития проекта (например, во время определения приоритетов невыполненной работы или при обзоре промежуточных этапов). Если изменения приняты, то следует использовать некоторую форму анализа прослеживаемости и анализа рисков, чтобы установить влияние этих изменений (см. раздел 2.5 «Управление рисками и контроль конфигурации программного обеспечения» в КА «Управление конфигурацией программного обеспечения»).

Подход управляемых изменений также может стать основой для оценки успеха во время закрытия поэтапного цикла или всего проекта на основе изменений, которые произошли в процессе (см. тему 5, Заккрытие).

2 Планирование программного проекта

Первым шагом в планировании проекта программного обеспечения должен быть выбор подходящей модели жизненного цикла разработки программного обеспечения и, возможно, ее адаптация на основе масштаба проекта, требований к программному обеспечению и оценки рисков. Другие факторы, которые необходимо учитывать, включают характер предметной области, функциональную и техническую сложность, а также требования к качеству программного обеспечения (см. Требования к качеству программного обеспечения в КА качества программного обеспечения).

Во всех SDLC оценка риска должна быть элементом начального планирования проекта, а «профиль риска» проекта должен обсуждаться и приниматься всеми соответствующими заинтересованными сторонами. Процессы управления качеством программного обеспечения (см. «Процессы управления качеством программного обеспечения» в КА «Качество программного обеспечения») должны быть определены как часть процесса планирования и привести к процедурам и обязанностям по обеспечению качества программного обеспечения, проверке и валидации, обзорам и аудитам (см. КА «Качество программного обеспечения»). Процессы и обязанности по текущему обзору и пересмотру плана проекта и связанных с ним планов также должны быть четко определены и согласованы.

2.1 Планирование процесса

[3 , c3, c4, c5] [5 , c1]

Модели жизненного цикла разработки программного обеспечения (SDLC) охватывают континуум от прогностического до адаптивного (см. Модели жизненного цикла программного обеспечения в процессе разработки программного обеспечения КА). Прогнозирующие SDLC характеризуются разработкой подробных требований к программному обеспечению, подробным планированием проекта и минимальным планированием итераций между этапами разработки. Адаптивные SDLC предназначены для удовлетворения возникающих требований к программному обеспечению и итеративной корректировки планов. SDLC с высоким уровнем прогнозирования выполняет первые пять процессов, перечисленных на рис. 7.1, в линейной последовательности с изменениями предыдущих фаз только по мере необходимости. Адаптивные SDLC характеризуются итеративными циклами разработки.

Хорошо известные SDLC включают водопадную, инкрементную и спиральную модели, а также различные формы гибкой разработки программного обеспечения [2] [3*, c2].

Соответствующие методы (см. Модели и методы разработки программного обеспечения КА) и инструменты должны быть выбраны как часть планирования. Автоматизированные инструменты, которые будут использоваться на протяжении всего проекта, также должны быть запланированы и приобретены. Инструменты могут включать инструменты для планирования проекта, требований к программному обеспечению, проектирования программного обеспечения, создания программного обеспечения, обслуживания программного обеспечения, управления конфигурацией программного обеспечения, процесса разработки программного обеспечения, качества программного обеспечения и другие. Хотя многие из этих инструментов следует выбирать, прежде всего, исходя из технических соображений, обсуждаемых в других КА, некоторые из них тесно связаны с управленческими соображениями, обсуждаемыми в этой главе.

2.2 Определение результатов

[3 , c4, c5, c6]

Рабочий продукт (продукты) каждой проектной деятельности (например, проектная документация по архитектуре программного обеспечения, отчеты об инспекциях, протестированное программное обеспечение) должен быть идентифицирован и охарактеризован. Следует оценить возможности повторного использования программных компонентов из предыдущих проектов или использования готовых программных продуктов. Необходимо планировать закупку программного обеспечения и привлекать третьих лиц для разработки результатов, а также выбирать поставщиков (см. раздел 3.2 «Приобретение программного обеспечения и управление контрактами с поставщиками»).

2.3 Усилия, график и оценка стоимости

[3 , с6]

Расчетный диапазон усилий, необходимых для проекта или частей проекта, можно определить с помощью откалиброванной модели оценки, основанной на исторических данных о размерах и усилиях (если они доступны) и других соответствующих методах, таких как экспертная оценка и аналогия. Можно установить зависимости задач, а потенциальные возможности для одновременного и последовательного выполнения задач можно определить и задокументировать, например, с помощью диаграммы Ганта. Для предиктивных проектов SDLC ожидаемое расписание задач с прогнозируемым временем начала, продолжительностью и временем окончания обычно составляется во время планирования. Для адаптивных проектов SDLC общая оценка усилий и графика обычно разрабатывается на основе первоначального понимания требований или, альтернативно,

Требования к ресурсам (например, люди и инструменты) могут быть переведены в смету затрат. Первоначальная оценка усилий, графика и стоимости — это повторяющаяся деятельность, которая должна обсуждаться и пересматриваться заинтересованными сторонами до тех пор, пока не будет достигнут консенсус в отношении ресурсов и времени, доступных для завершения проекта.

2.4 Распределение ресурсов

[3 , с5, с10, с11]

Оборудование, средства и люди должны быть распределены по определенным задачам, включая распределение ответственности за выполнение различных элементов проекта и проекта в целом. Можно использовать матрицу, которая показывает, кто несет ответственность, подотчетен, с кем консультируются и информируются о каждой из задач. Распределение ресурсов основано и ограничено наличием ресурсов и их оптимальным использованием, а также проблемами, связанными с персоналом (например, производительностью отдельных лиц и команд, динамикой команды и структурой команды).

2.5 Управление рисками

[3 , с9][5 , с5]

Риск и неопределенность являются связанными, но разными понятиями. Неопределенность возникает из-за недостатка информации. Риск характеризуется вероятностью события, которое приведет к негативному воздействию, а также характеристикой негативного воздействия на проект. Риск часто является результатом неопределенности. Обратной стороной риска является возможность, которая характеризуется вероятностью того, что событие, имеющее положительный исход, может произойти.

Управление рисками включает в себя идентификацию факторов риска и анализ вероятности и потенциального воздействия каждого фактора риска, приоритизацию факторов риска и разработку стратегий снижения риска для снижения вероятности и минимизации негативного воздействия, если фактор риска становится проблемой. Методы оценки риска (например, экспертная оценка, исторические данные, деревья решений и моделирование процессов) иногда могут использоваться для выявления и оценки факторов риска.

Условия отказа от проекта также могут быть определены на этом этапе в ходе обсуждения со всеми соответствующими заинтересованными сторонами. Аспекты риска, характерные только для программного обеспечения, такие как склонность инженеров-программистов добавлять ненужные функции или риски, связанные с нематериальным характером программного обеспечения, могут влиять на управление рисками программного проекта. Особое внимание следует уделить управлению рисками, связанными с требованиями к качеству программного обеспечения, такими как безопасность или

защищенность (см. КА качества программного обеспечения). Управление рисками должно осуществляться не только в начале проекта, но и через определенные промежутки времени на протяжении всего жизненного цикла проекта.

2.6 Управление качеством

[3 , с4][4 , с24]

Требования к качеству программного обеспечения должны быть определены, возможно, как в количественном, так и в качественном отношении, для программного проекта и связанных с ним рабочих продуктов. Пороги для приемлемых измерений качества должны быть установлены для каждого требования к качеству программного обеспечения на основе потребностей и ожиданий заинтересованных сторон. Процедуры, связанные с текущим обеспечением качества программного обеспечения (SQA) и улучшением качества на протяжении всего процесса разработки, а также для проверки и проверки готового программного продукта, также должны быть указаны во время планирования качества (например, технические обзоры и проверки или демонстрации завершенной функциональности; см. КА качества программного обеспечения).

2.7 Управление планом

[3 , с4]

Для программных проектов, где ожидаются изменения, необходимо управлять планами. Таким образом, управление планом проекта должно быть запланировано. Планы и процессы, выбранные для разработки программного обеспечения, должны систематически отслеживаться, пересматриваться, сообщаться и, при необходимости, пересматриваться. Планы, связанные с вспомогательными процессами (например, документация, управление конфигурацией программного обеспечения и решение проблем), также должны управляться. Отчетность, мониторинг и управление проектом должны соответствовать выбранному SDLC и реалиям проекта; планы должны учитывать различные артефакты, которые будут использоваться для управления проектом.

3 Принятие решения о программном проекте

Во время реализации программного проекта (также известного как выполнение проекта) реализуются планы и реализуются процессы, воплощенные в планах. На протяжении всего процесса следует уделять особое внимание соблюдению выбранных процессов SDLC с преобладающим ожиданием того, что соблюдение приведет к успешному удовлетворению требований заинтересованных сторон и достижению целей проекта. Основой принятия являются непрерывные управленческие действия по мониторингу, контролю и отчетности.

3.1 Реализация планов

[4 , с2]

Деятельность по проекту должна осуществляться в соответствии с планом проекта и вспомогательными планами. Ресурсы (например, персонал, технологии и финансирование) используются и создаются рабочие продукты (например, разработка программного обеспечения, программный код и тестовые примеры программного обеспечения).

3.2 Приобретение программного обеспечения и управление контрактами с поставщиками

[3 , с3, с4]

Приобретение программного обеспечения и управление контрактами с поставщиками связаны с вопросами, связанными с заключением контрактов с клиентами организации-разработчика программного обеспечения, которые приобретают предоставляемые рабочие продукты, и с поставщиками, которые поставляют продукты или услуги организации, занимающейся разработкой программного обеспечения.

Это может включать выбор соответствующих видов контрактов, таких как фиксированная цена, время и материалы, стоимость плюс фиксированная плата или стоимость плюс поощрительное вознаграждение. В соглашениях с клиентами и поставщиками обычно указывается объем работ и результаты, а также такие положения, как штрафы за несвоевременную поставку или недопоставку, а также соглашения об интеллектуальной собственности, в которых указывается, что поставщик или поставщики предоставляют

и за что платит приобретатель, а также то, что будет передано приобретателю и принадлежат ему. Для программного обеспечения, разрабатываемого поставщиками (как внутренними, так и внешними по отношению к организации-разработчику программного обеспечения), в соглашениях обычно указываются требования к качеству программного обеспечения для приемки поставленного программного обеспечения.

После заключения соглашения необходимо управлять выполнением проекта в соответствии с условиями соглашения (дополнительную информацию по этой теме см. в главе 12 SWX «Управление закупками программного обеспечения» [2]).

3.3 Реализация процесса измерения

[3 , с7]

Процесс измерения должен быть реализован во время проекта программного обеспечения, чтобы обеспечить сбор релевантных и полезных данных (см. разделы 6.2 «Планирование процесса измерения» и 6.3 «Выполнение процесса измерения»).

3.4 Мониторинг процесса

[3 , с8]

Соблюдение плана проекта и связанных с ним планов следует оценивать постоянно и через заранее установленные промежутки времени. Кроме того, должны быть оценены результаты и критерии завершения для каждой задачи. Результаты должны оцениваться с точки зрения их требуемых характеристик (например, посредством проверок или демонстрации рабочих функций). Следует проанализировать затраты усилий, соблюдение графика и текущие затраты, а также изучить использование ресурсов. Следует пересмотреть профиль рисков проекта (см. раздел 2.5 «Управление рисками») и оценить соблюдение требований к качеству программного обеспечения (см. «Требования к качеству программного обеспечения» в КА «Качество программного обеспечения»).

Данные измерений должны быть проанализированы (см. Статистический анализ в Основах проектирования). Должен быть определен дисперсионный анализ, основанный на отклонении фактических результатов и значений от ожидаемых. Это может включать перерасход средств, отставание от графика или другие подобные меры. Должны быть выполнены выявление выбросов и анализ качества и других данных измерений (например, анализ дефектов; см. Измерение качества программного обеспечения в КА качества программного обеспечения). Подверженность риску должна быть пересчитана (см. раздел 2.5 «Управление рисками»). Эти действия могут обеспечить обнаружение проблем и идентификацию исключений на основе превышения пороговых значений. О результатах следует сообщать в случае превышения пороговых значений или по мере необходимости.

3.5 Процесс управления

[3 , с7, с8]

Результаты деятельности по мониторингу проекта обеспечивают основу для принятия решений. Там, где это уместно, и когда вероятность и влияние факторов риска понятны, в проект могут быть внесены изменения. Это может принимать форму корректирующих действий (например, повторное тестирование определенных компонентов программного обеспечения); это может включать в себя включение дополнительных действий (например, принятие решения об использовании прототипирования для помощи в проверке требований к программному обеспечению; см. Прототипирование в требованиях к программному обеспечению КА); и/или это может повлечь за собой пересмотр плана проекта и других проектных документов (например, спецификацию требований к программному обеспечению) для учета непредвиденных событий и их последствий.

В некоторых случаях процесс контроля может привести к отказу от проекта. Во всех случаях необходимо соблюдать процедуры контроля конфигурации программного обеспечения и управления конфигурацией программного обеспечения (см. КА «Управление конфигурацией программного обеспечения»), решения следует документировать и сообщать всем соответствующим сторонам, планы следует пересматривать и пересматривать при необходимости, а соответствующие данные регистрировать (см. раздел 6.3, Выполнение процесса измерения).

3.6 Отчетность

[3 , с11]

В указанные и согласованные сроки следует сообщать о достигнутом прогрессе как внутри организации (например, руководящему комитету проекта), так и внешним заинтересованным сторонам (например, клиентам или пользователям). Отчеты должны быть сосредоточены на информационных потребностях целевой аудитории, а не на подробных отчетах о состоянии внутри проектной группы.

4 Обзор и оценка

В заранее установленное время и по мере необходимости следует оценивать общий прогресс в достижении заявленных целей и удовлетворении требований заинтересованных сторон (пользователей и клиентов). Точно так же следует регулярно и в соответствии с обстоятельствами проводить оценку эффективности программного процесса, задействованного персонала, а также используемых инструментов и методов.

4.1 Определение удовлетворения требований

[4 , с8]

Поскольку достижение удовлетворенности заинтересованных сторон является основной целью менеджера по разработке программного обеспечения, следует периодически оценивать прогресс в достижении этой цели. Прогресс следует оценивать по достижению основных вех проекта (например, завершение проектирования архитектуры программного обеспечения или завершение технического обзора программного обеспечения) или по завершении итеративного цикла разработки, результатом которого является приращение продукта. Следует выявлять отклонения от требований к программному обеспечению и предпринимать соответствующие действия.

Как и в описанном выше процессе управления (см. раздел 3.5 «Процесс управления»), следует соблюдать процедуры контроля конфигурации программного обеспечения и управления конфигурацией программного обеспечения (см. КА «Управление конфигурацией программного обеспечения»), решения документировать и сообщать всем соответствующим сторонам, планы пересматривать и пересматривать, если необходимы, и соответствующие данные записаны (см. раздел 6.3, Выполнение процесса измерения).

4.2 Обзор и оценка производительности

[3 , с8, с10]

Периодические обзоры производительности персонала проекта могут дать представление о вероятности соблюдения планов и процессов, а также о возможных трудностях (например, конфликты между членами команды). Различные используемые методы, инструменты и приемы следует оценивать на предмет их эффективности и уместности, а процесс, используемый в проекте, также следует систематически и периодически оценивать на предмет актуальности, полезности и действенности в контексте проекта. Там, где это уместно, следует вносить изменения и управлять ими.

5 Заккрытие

Весь проект, основная фаза проекта или цикл итеративной разработки завершается, когда все планы и процессы приняты и завершены. Должны быть оценены критерии успеха проекта, этапа или итерации. Как только закрытие установлено, можно выполнять действия по архивированию, ретроспективе и совершенствованию процессов.

5.1 Определение закрытия

[1 , с3.7, с4.6]

Закрытие происходит, когда указанные задачи для проекта, фазы или итерации выполнены и подтверждено удовлетворительное достижение критериев завершения. Требования к программному обеспечению могут быть подтверждены как удовлетворенные или неудовлетворенные, и может быть определена степень достижения целей. В процессах закрытия должны участвовать соответствующие заинтересованные стороны, и в результате должно быть документально подтверждено согласие соответствующих заинтересованных сторон; любые известные проблемы должны быть задокументированы.

5.2 Действия по закрытию

[2 , с3.7, с4.8]

После подтверждения закрытия следует выполнить архивирование материалов проекта в соответствии с согласованными с заинтересованными сторонами методами, местом и продолжительностью, возможно, включая уничтожение конфиденциальной информации, программного обеспечения и носителя, на котором хранятся копии. База данных измерений организации должна быть обновлена соответствующими проектными данными. Следует провести ретроспективный анализ проекта, фазы или итерации, чтобы можно было проанализировать возникшие вопросы, проблемы, риски и возможности (см. тему 4 «Обзор и оценка»). Уроки, извлеченные из проекта, следует использовать для организационного обучения и улучшения.

6 Измерение программной инженерии

Важность измерения и его роль в улучшении управления и инженерных методов широко признаны (см. Измерение в Основах инженерии КА). Эффективное измерение стало одним из краеугольных камней организационной зрелости. Измерение может применяться к организациям, проектам, процессам и рабочим продуктам. В этом разделе основное внимание уделяется применению измерений на уровнях проектов, процессов и рабочих продуктов.

Этот раздел соответствует стандарту IEEE 15939:2008 [6], который описывает процесс определения действий и задач, необходимых для реализации процесса измерения программного обеспечения. Стандарт также включает информационную модель измерений.

6.1 Установление и поддержание обязательств по измерению

[7 , с1, с2]

- Требования к измерению. Каждая попытка измерения должна руководствоваться

целей организации и определяется набором требований к измерениям, установленных организацией и проектом (например, целью организации может быть «первым вывести на рынок новые продукты»).

- Объем измерения. Должна быть установлена организационная единица, к которой должно применяться каждое требование измерения. Это может состоять из функциональной области, одного проекта, одного сайта или всего предприятия. Следует также учитывать временные рамки усилий по измерению, поскольку

могут потребоваться временные ряды некоторых измерений; например, для калибровки моделей оценки (см. раздел 2.3, Оценка усилий, графика и затрат).

- Приверженность команды измерениям. Обязательство должно быть официально установлено, доведено до сведения и подкреплено ресурсами (см. следующий пункт).
- Ресурсы для измерения. Приверженность организации измерениям

существенный фактор успеха, о чем свидетельствует выделение ресурсов для осуществления процесса измерения. Назначение ресурсов включает в себя распределение ответственности за различные задачи процесса измерения (например, аналитик и библиотекарь). Также должны быть выделены адекватное финансирование, обучение, инструменты и поддержка для проведения процесса.

6.2 Планирование процесса измерения

[7 , с1, с2]

- Охарактеризуйте организационную единицу. Организационная единица обеспечивает контекст для измерения, поэтому организационный контекст должен быть явным, включая ограничения, которые организация накладывает на процесс измерения. Характеристика может быть сформулирована с точки зрения организационной

процессы, домены приложений, технологии, организационные интерфейсы и организационная структура.

- Выявить потребности в информации. Информационные потребности основаны на целях, ограничениях, рисках и проблемах организационной единицы. Они могут быть получены из

деловых, организационных, нормативных и/или продуктовых

цели. Они должны быть определены и расставлены по приоритетам. Затем подмножество задач, которые необходимо решить, может быть выбрано, задокументировано, доведено до сведения и рассмотрено заинтересованными сторонами.

- Выберите меры. Следует выбирать меры-кандидаты с четкой связью с информационными потребностями. Меры следует выбирать на основе приоритетов информационных потребностей и других критериев, таких как стоимость сбора,

степень нарушения процесса во время сбора данных, простота получения точных, непротиворечивых данных, а также простота анализа и отчетности. Поскольку внутренние характеристики качества (см. Модели и характеристики качества в КА качества программного обеспечения) часто не содержатся в юридически обязывающих требованиях к программному обеспечению, важно рассмотреть возможность измерения внутреннего качества программного обеспечения, чтобы обеспечить ранний индикатор потенциальных проблем, которые могут повлиять на внешние заинтересованные стороны.

- Определить процедуры сбора, анализа и отчетности. Это включает в себя процедуры сбора и графики, хранение, проверку, анализ, отчетность и управление конфигурацией данных.
- Выберите критерии оценки информационных продуктов. На критерии оценки влияют технические и бизнес-цели организационной единицы. Информационные продукты включают продукты, связанные с

производимого продукта, а также те, которые связаны с процессами, используемыми для управления и измерения проекта.

- Предоставьте ресурсы для задач измерения. План измерений должен быть рассмотрен и одобрен соответствующими заинтересованными сторонами, чтобы включить все процедуры сбора данных; процедуры хранения, анализа и отчетности; оценка

критерии; расписания; и обязанности. Критерии проверки этих артефактов должны быть установлены на уровне организационного подразделения или выше и должны использоваться в качестве основы для этих проверок. Такие критерии должны учитывать предыдущий опыт, наличие ресурсов и потенциальные сбои в проектах, когда предлагаются изменения по сравнению с текущей практикой. Утверждение демонстрирует приверженность процессу измерения.

- Определите ресурсы, которые должны быть доступны для реализации запланированных и утвержденных задач измерения. Доступность ресурсов может быть поэтапной в тех случаях, когда изменения должны быть опробованы перед массовым развертыванием. Следует уделить внимание ресурсам, необходимым для успешного развертывания новых

процедуры или меры.

- Приобретайте и внедряйте вспомогательные технологии. Это включает в себя оценку имеющихся вспомогательных технологий, выбор наиболее подходящих

технологии, приобретение этих технологий и развертывание этих технологий.

6.3 Выполнение процесса измерения

[7 , c1, c2]

- Интегрируйте процедуры измерения с соответствующими программными процессами. Процедуры измерения, такие как сбор данных, должны быть интегрированы в программные процессы, которые они измеряют. Это может включать изменение

текущие программные процессы для обеспечения деятельности по сбору или генерации данных. Это может также включать анализ текущих программных процессов, чтобы свести к минимуму дополнительные усилия, и оценку воздействия на сотрудников, чтобы убедиться, что процедуры измерения будут приняты. Следует учитывать вопросы морали и другие человеческие факторы. Кроме

того, процедуры измерения должны быть доведены до сведения тех, кто предоставляет данные. Также может потребоваться обучение и поддержка. Процедуры анализа данных и отчетности обычно сходным образом интегрируются в организационные и/или проектные процессы.

- Собирать данные. Данные должны быть собраны, проверены и сохранены. Сбор иногда можно автоматизировать с помощью инструментов управления разработкой программного обеспечения (см. раздел 7, Инструменты управления разработкой программного обеспечения) для анализа данных и разработки отчетов. Данные могут быть агрегированы, преобразованы или перекодированы как часть процесса анализа с использованием степени строгости, соответствующей характеру данных.

и потребности в информации. Результаты этого анализа обычно представляют собой индикаторы, такие как графики, числа или другие признаки, которые будут интерпретироваться, что приведет к выводам и рекомендациям, которые будут представлены заинтересованным сторонам (см. Статистический анализ в Инженерных основах КА). Результаты и выводы обычно анализируются с использованием процесса, определенного организацией (который может быть формальным или неформальным). Поставщики данных и пользователи измерений должны участвовать в рассмотрении данных, чтобы убедиться, что они содержательны и точны и что они могут привести к разумным действиям.

- Сообщите результаты. Информационные продукты должны быть задокументированы и доведены до сведения пользователей и заинтересованных сторон.

6.4 Оценка измерения

[7 , с1, с2]

- Оценивать информационные продукты и процесс измерения по установленным критериям оценки и определять сильные и слабые стороны информационных продуктов или процесса соответственно. Оценка может быть выполнена внутренним процессом или внешним аудитом; он должен включать обратную связь от пользователей измерений. Извлеченные уроки должны быть зарегистрированы в соответствующей базе данных.
- Определите потенциальные улучшения. Такими улучшениями могут быть изменения в формате показателей, изменения в единицах измерения или реклассификация категорий измерения. Должны быть определены затраты и выгоды от потенциальных улучшений, и должны быть представлены соответствующие действия по улучшению.
- Сообщите о предлагаемых улучшениях владельцу процесса измерения и заинтересованным сторонам для рассмотрения и утверждения. Кроме того, следует сообщать об отсутствии потенциальных улучшений, если анализ не выявил каких-либо улучшений.

7 инструментов управления программной инженерией

[3 , с5, с6, с7]

Инструменты управления программной инженерией часто используются для обеспечения видимости и контроля над процессами управления программной инженерией. Некоторые инструменты автоматизированы, а другие реализуются вручную. В последнее время наметилась тенденция к использованию интегрированных наборов инструментов разработки программного обеспечения, которые используются на протяжении всего проекта для планирования, сбора и записи, мониторинга и контроля, а также отчетности по проектам и продуктам. Инструменты можно разделить на следующие категории:

Инструменты планирования и отслеживания проекта. Инструменты планирования и отслеживания проекта можно использовать для оценки трудозатрат и стоимости проекта, а также для подготовки графиков проекта. В некоторых проектах используются автоматизированные инструменты оценки, которые принимают в качестве входных данных предполагаемый размер и другие характеристики программного продукта и производят оценки требуемых общих усилий, графика и стоимости. Инструменты планирования также включают инструменты автоматизированного планирования, которые анализируют задачи в рамках структурной декомпозиции работ, их предполагаемую продолжительность, их отношения приоритета и ресурсы, назначенные каждой задаче, для создания расписания в форме диаграммы Ганта.

Инструменты отслеживания можно использовать для отслеживания этапов проекта, регулярно запланированных совещаний по статусу проекта, запланированных циклов итераций, демонстраций продуктов и/или действий.

Инструменты управления рисками. Инструменты управления рисками (см. раздел 2.5 «Управление рисками») могут использоваться для отслеживания выявления, оценки и мониторинга рисков. Эти инструменты включают использование таких подходов, как моделирование или деревья решений, для анализа влияния затрат на выигрыши и субъективные оценки вероятностей событий риска. Инструменты моделирования методом Монте-Карло можно использовать для создания вероятностных распределений усилий, расписания и риска путем объединения нескольких входных вероятностных распределений алгоритмическим способом.

Средства связи. Коммуникационные средства могут помочь в предоставлении своевременной и последовательной информации соответствующим заинтересованным сторонам, участвующим в проекте. Эти инструменты могут включать в себя такие вещи, как уведомления по электронной почте и трансляции для членов команды и заинтересованных сторон. Они также включают протоколы регулярно запланированных совещаний по проекту, ежедневных стендовых совещаний, а также диаграммы, показывающие прогресс, незавершенные работы и решения по запросам на техническое обслуживание.

Инструменты измерения. Инструменты измерения поддерживают действия, связанные с программой измерения программного обеспечения (см. раздел 6, Измерение программной инженерии). В этой категории есть несколько полностью автоматизированных инструментов. Инструменты измерения, используемые для сбора, анализа и представления данных измерений проекта, могут быть основаны на электронных таблицах, разработанных членами проектной группы или сотрудниками организации.

ДАЛЬНЕЙШИЕ ЧТЕНИЯ

Стивен П. Берчук и Брэд Эпплтон, *Шаблоны управления конфигурацией программного обеспечения: эффективная командная работа, практическая интеграция* [6].

В этой книге полезные практики и стратегии SCM представлены в виде шаблонов. Шаблоны могут быть реализованы с использованием различных инструментов, но они выражаются независимым от инструментов способом.

«СММІ для разработки», версия 1.3, стр. 137–147 [7].

Эта модель представляет собой набор лучших практик, которые помогут организациям, занимающимся разработкой программного обеспечения, улучшить свои процессы. На уровне зрелости 2 он предлагает действия по управлению конфигурацией.

ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА

- [1] *ISO/IEC/IEEE 24765:2010 Системная и программная инженерия. Словарь*, ISO/IEC/IEEE, 2010.
- [2] *Стандарт IEEE. 828-2012, Стандарт управления конфигурацией в системной и программной инженерии*, IEEE, 2012.
- [3] AMJ Nass, *Принципы и практика управления конфигурацией*, 1-е изд., Addison Wesley, 2003.
- [5] И. Соммервилль, *Программная инженерия*, 9-е изд., 9-е изд., Аддисон Уэсли, 2011.
- [6] С.П. Берчук и Б. Эпплтон, *Шаблоны управления конфигурацией программного обеспечения: эффективная командная работа, практическая интеграция*, Addison Wesley Professional, 2003.
- [7] Команда разработчиков СММІ, «СММІ для разработки, версия 1.3», Институт программной инженерии, 2010 г., <http://resources.sei.cmu.edu/library/asset-view.cfm?assetID=9661>.

Retrieved from "http://swbokwiki.org/index.php?title=Chapter_7:_Software_Engineering_Management&oldid=532"

-
- Последнее изменение этой страницы состоялось 25 августа 2015 г., в 16:17.