

**Пигаль Анастасия Сергеевна**, ассистент кафедры ВМиТ, DoroshenkoAS@yandex.ru

**Пигаль Павел Борисович**, ассистент кафедры ВМиТ

Полесский государственный университет

**Аннотация:** описаны условия и критерии эффективности методологий управления проектами в применении к информационным технологиям.

**Ключевые слова:** методология, информационные системы, оптимизация задач, программирование.

Введение. Определение методологии управления проектом в современной ИТ-сфере является одним из важнейших требований к выполнению задач. От того какой путь, какую методологию вы выберете, зависит напрямую какой результат вы получите и как скоро вы его получите. Можно выделить несколько свойств всех процессов, по которым можно судить насколько эффективен данный метод в конкретной ситуации.

1. Скорость реализации проекта.
2. Расходы на создание проекта.
3. Управление рисками.
4. Время до начала проекта.

Понятие *скорости реализации проекта* – это время, затраченное на весь проект в целом. Начиная с обсуждения выбора методологии по которой этот проект будет реализовываться и заканчивая готовым продуктом.

*Расходы на создание проекта* – эта величина выражается в денежном эквиваленте и в целом охватывает все ресурсы (расходы) затраченные на создание проекта на протяжении всего времени, включая издержки связанные с рисками.

*Управление рисками.* В настоящее время управление информационными рисками представляет собой одно из наиболее актуальных и динамично развивающихся направлений стратегического и оперативного менеджмента в области защиты информации. Его основная задача — объективно идентифицировать и оценить наиболее значимые для бизнеса информационные риски компании, а также адекватность используемых средств контроля рисков для увеличения эффективности и рентабельности экономической деятельности компании. Поэтому под термином «управление информационными рисками» обычно понимается системный процесс идентификации, контроля и уменьшения информационных рисков компаний в соответствии с определенными ограничениями нормативно-правовой базы в области защиты информации и собственной корпоративной политики безопасности. Считается, что качественное управление рисками позволяет использовать оптимальные по эффективности и затратам средства контроля рисков и средства защиты информации, адекватные текущим целям и задачам бизнеса компании.[1]

*Время до начала проекта* – это время, затраченное на предпроектную подготовку. Создание документации, описание бизнес-процессов и т.п.

## PMI PMBOK5

Наиболее распространенной методологией управления проектами в сфере информационных технологий является методология *PMI PMBOK5* Американского института управления проектами – PMI.

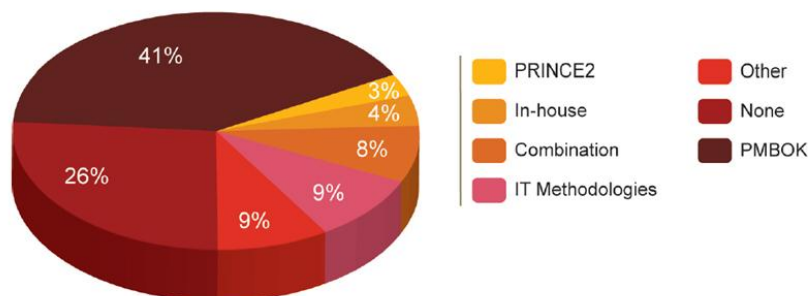


Рисунок – Распространенность использования методологий.

Основной плюс данного процесса заключается в том, что в его основу положен процессный подход. Весь алгоритм управления любым проектом разделен на 5 групп процессов (инициализация, планирование, исполнение, мониторинг и контроль и завершение), которые в свою очередь разделены на 47 процессов.[2]

По *скорости реализации проекта* данная методология проигрывает всем остальным т.к. она рассчитана на проекты от 3 лет длительности.

По *расходам на создание проекта*. Минусом является громоздкость и большие затраты на планирование и разработку проектной документации.

По *управлению рисками*. Из-за низкой скорости реализации проекта проект может быть “свернут” еще на стадии разработки проектной документации, например из-за возникновения больших затрат на содержание проекта.

По *времени до начала проекта*. Очень много времени затрачивается на планирование и разработку проектной документации, зачастую даже больше чем на реализацию самого проекта.

### PRINCE2

Стандарт PRINCE2 был разработан в 1989 году по заказу Британского агентства Central Computer and Telecommunications Agency (CCTA). В 1996 году он был принят в качестве основного стандарта в области управления проектами в Великобритании. Согласно исследованию, проведенному PWC (рис. 1) в мире методологию PRINCE2 использует порядка 3% компаний.

Практическая польза данного стандарта заключается в том, что он даёт пошаговый алгоритм управления проектом и сфокусирован на достижении бизнес-задач, ради которых он был принят. По выделенным нами свойствам методология **PRINCE2** схожа с **PMI PMBOK5**, но более ориентирована для ИТ-индустрии. Скорость разработки предпроектной документации уменьшено, соответственно уменьшаются риски, и скорость разработки самого проекта уменьшена.

### AGILE

Данное семейство методологий гибкой разработки проектов использует порядка 9% организаций. Самой известной методологией данного семейства в настоящий момент является SCRUM. [3]

Методология Agile была разработана для управления ИТ-проектами в области разработки программного обеспечения и базируется на принципах противоположных классическому подходу, который проповедует PMBOK.

Основные принципы методологии Agile:

- Короткий итеративный цикл разработки. (1-6 недель)
- Фокус на продукте, а не проектной документации.
- Фокус на коммуникациях в команде.
- Гибкий процесс внесения изменения в проект.
- Наличие владельца продукта – заказчика, который определяет требования к продукту.

На первом месте методология Agile ставит участников проекта и их взаимодействие, затем уже рассматривают продукт, после сотрудничество с заказчиком, реакцию на изменения.[4]

По *скорости реализации проекта* – проект может быть затянут из-за большого количества итераций (в конечном итоге может быть “свернут” как не рентабельный).

По *расходам на создание проекта* – непредсказуемый бюджет разработки.

По *управлению рисками* – имеет значительные риски: затягивание проекта, потеря актуальности решаемой задачи и т.п.

По *времени до начала проекта*. Предпроектной подготовки не требует. Работа над проектом может начаться как только соберутся все участники проекта.

### **MSF**

Microsoft Solutions Framework (MSF) – это методология разработки программного обеспечения, созданная корпорацией Microsoft в 1994 году на основе своего многолетнего опыта работы в ИТ-индустрии.

По сути, данная методология входит в семейство Agile-методов гибкой разработки программного обеспечения.

Данная методология состоит из двух взаимосвязанных framework'ов: Microsoft Solutions Framework (MSF) и Microsoft Operations Framework (MOF).

Данная модель состоит из 5 процессов управления проектами:

- Выработка концепции (Envisioning)
- Планирование (Planning)
- Разработка (Developing)
- Стабилизация (Stabilizing)
- Внедрение (Deploying)

### **RUP**

В основе RUP лежат следующие принципы:

- Ранняя идентификация и непрерывное (до окончания проекта) устранение основных рисков.
- Концентрация на выполнении требований заказчиков к исполняемой программе (анализ и построение модели прецедентов (вариантов использования)).
- Ожидание изменений в требованиях, проектных решениях и реализации в процессе разработки.
- Компонентная архитектура, реализуемая и тестируемая на ранних стадиях проекта.
- Постоянное обеспечение качества на всех этапах разработки проекта (продукта).
- Работа над проектом в сплочённой команде, ключевая роль в которой принадлежит архитекторам.

RUP использует итеративную модель разработки. В конце каждой итерации (в идеале продолжающейся от 2 до 6 недель) проектная команда должна достичь запланированных на данную итерацию целей, создать или доработать проектные артефакты и получить промежуточную, но функциональную версию конечного продукта. Итеративная разработка позволяет быстро реагировать на меняющиеся требования, обнаруживать и устранять риски на ранних стадиях проекта, а также эффективно контролировать качество создаваемого продукта.

Полный жизненный цикл разработки продукта состоит из четырех фаз, каждая из которых включает в себя одну или несколько итераций:

#### 1. Начальная стадия (Inception)]

В фазе начальной стадии:

Формируются видение и границы проекта. Создается экономическое обоснование (business case). Определяются основные требования, ограничения и ключевая функциональность продукта. Создается базовая версия модели прецедентов. Оцениваются риски.

При завершении начальной фазы оценивается достижение вехи целей жизненного цикла (англ. Lifecycle Objective Milestone), которое предполагает соглашение заинтересованных сторон о продолжении проекта.

#### 2. Уточнение (Elaboration)

В фазе «Уточнение» производится анализ предметной области и построение исполняемой архитектуры. Это включает в себя:

Документирование требований (включая детальное описание для большинства прецедентов). Спроектированную, реализованную и оттестированную исполняемую архитектуру. Обновленное экономическое обоснование и более точные оценки сроков и стоимости. Сниженные основные риски.

Успешное выполнение фазы разработки означает достижение вехи архитектуры жизненного цикла (англ. Lifecycle Architecture Milestone).

### 3. Построение (Construction)

В фазе «Построение» происходит реализация большей части функциональности продукта. Фаза Построение завершается первым внешним релизом системы и вехой начальной функциональной готовности (Initial Operational Capability).

### 4. Внедрение (Transition)

В фазе «Внедрение» создается финальная версия продукта и передается от разработчика к заказчику. Это включает в себя программу бета-тестирования, обучение пользователей, а также определение качества продукта. В случае, если качество не соответствует ожиданиям пользователей или критериям, установленным в фазе Начало, фаза Внедрение повторяется снова. Выполнение всех целей означает достижение вехи готового продукта (Product Release) и завершение полного цикла разработки [5].

В данный момент в ИТ-индустрии наиболее распространенным является комплексный подход к решению поставленных задач. Т.е. используется смесь методологий управления проектами. При использовании конкретных методологий компании разработчики привлекают специалистов в конкретной области методологии. Все больше набирает популярность методология **SCRUM**, по причине того, что SCRUM позволяет решать задачи при правильном подходе за короткий период времени, а также внедрять по ходу разработки различные новые элементы, заранее не предусмотренные проектом. Если смотреть со стороны разработки банковских систем наиболее удобной и понятной в данном случае будет методология **PMI PMBOK5**, или ее конкретная модель в ИТ-сфере waterfall.

### Список использованных источников:

1. <http://citforum.ru/security/articles/risk/>
2. <http://blog.pa-resultant.ru/archives/1070#.VEWdIvmsU71>
3. <https://ru.wikipedia.org/wiki/Scrum>
4. <http://www.slideshare.net/mikhailsofonov/agile-scrum-16980133>
5. [https://ru.wikipedia.org/wiki/Rational\\_Unified\\_Process](https://ru.wikipedia.org/wiki/Rational_Unified_Process)