

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Шебзухова Татьяна Александровна

Должность: Директор Пятигорского института (филиал) Северо-Кавказского
федерального университета

Дата подписания: 10.11.2023 12:24:03

Уникальный программный ключ:

d74ce93cd40e39275c3ba2f58486412a1c8ef96f

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ

Методические указания

по выполнению практических работ

по дисциплине

«ПРОЕКТНЫЙ МЕНЕДЖМЕНТ В РЕШЕНИИ ИНЖЕНЕРНЫХ ЗАДАЧ»

для направления подготовки **08.03.01 Строительство**

направленность (профиль) **Городское строительство и хозяйство**

**Пятигорск
2021**

СОДЕРЖАНИЕ

<u>ВВЕДЕНИЕ</u>	3
<u>1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ</u>	4
<u>2. ОБОРУДОВАНИЕ И МАТЕРИАЛЫ</u>	4
<u>3. Наименование практических занятий</u>	4
<u>СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКИХ ЗАНЯТИЙ</u>	5
<u>Практическая работа 1. Введение в проектный менеджмент</u>	5
<u>Практическая работа 2. Методы и средства проектного менеджмента</u>	12
<u>Практическая работа 3. Менеджмент этапов жизненного цикла инженерного проекта</u>	Ошибка! Закладка не определена.
<u>Практическая работа 4. Технологии проектного менеджмента в решении инженерных задач</u>	Ошибка! Закладка не определена.
<u>Практическая работа 5. Методология проектного менеджмента инженерных задач</u>	Ошибка! Закладка не определена.
<u>Практическая работа 6. Методы управления качеством инженерного проекта</u>	Ошибка! Закладка не определена.
<u>Практическая работа 7. Обзор современных технологий менеджмента инженерных задач</u>	
<u>Практическая работа 8. Гибкая методология управления проектами: Agile, Scrum, Kanban, XP, APF</u>	Ошибка! Закладка не определена.
<u>4. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ</u>	16
<u>4.1. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины</u>	16
<u>4.1.1. Перечень основной литературы</u>	16
<u>4.1.2. Перечень дополнительной литературы</u>	Ошибка! Закладка не определена.
<u>4.2. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине</u>	17
<u>4.3. Перечень ресурсов информационно-телекоммуникационной сети Интернет, необходимых для освоения дисциплины</u>	Ошибка! Закладка не определена.

ВВЕДЕНИЕ

В методических указаниях содержатся материалы, необходимые для самостоятельной подготовки студентов к выполнению практических работ. В описание практических работ включены цель работы, порядок ее выполнения, рассмотрены теоретические вопросы, связанные с реализацией поставленных задач, приведена необходимая литература.

Методические указания посвящены курсу «Проектный менеджмент в решении инженерных задач». Управлять людьми сложно. Управлять людьми эффективно — невероятно сложно. Поэтому мы изобретаем и смешиваем разные практики, алгоритмы и технологии контроля, оценки результатов труда. Переходим от устаревших методик к новым, стараемся учесть все нюансы и риски, тратим время и ресурсы на планирование вместо работы по накатанной.

Из множества методологий нужна одна оптимальная и настроенная лично под нас и наш проект. Одно дело выполнять шаблонные задачи строго по скриптам из книги, другое — методом проб и ошибок вырабатывать свои специальные инструменты для учета приоритетов компании. Поэтому проектный менеджмент (РМ), то есть методология управления компанией с делением всей работы на проекты, становится популярным во всех отраслях.

Многие компании только сейчас переходят от классической (отработанная, часто бюрократическая схема) к проектной (каждая задача отдельно, делегирование ответственности) модели управления. Общий менеджмент для предпринимателя сводился к тому, что есть руководитель и исполнитель задачи. Дисциплина взаимодействия зависела от того, как прописано в шаблоне. Шаблон же был вбит в разумы всех одинаково лет этак 20-30 назад. Только этот шаблон уже не применить к новым условиям мирового рынка.

Исторически все началось с того, что в период перестройки 90-х годов сама логика ведения бизнеса была нарушена настолько, что попытки построить новую рабочую концепцию управления в миллениум создали термин «компания-однодневка». Концепции таких компаний проверялись на прочность и только одна из 10 компаний существовала больше полугода в 2000-х. Параллельно оставались многолетние предприятия, кардинально меняясь каждые лет 5 чтобы выжить в период перемен.

В 2010-х информационный взрыв интернета сделал доступной всю разрозненную базу наработок европейского и американского бизнеса. Из тонн полезной и мусорной информации вроде «что такое управление проектами и современный менеджмент», «как распределять обязанности, правильно предугадывать и сокращать риски» предприниматели до сих пор выбирают крупинки знаний, применимых именно в их компаниях.

Сейчас мы следим в новостях как прибыльные корпорации растут и распадаются на подразделения, покупаются и переживают структурные слияния. За каждым актом купли-продажи отдела кроется сложная сеть проектов (гаджеты, приложения, расширения для браузеров), связанных по разным критериям. Когда меняются критерии, меняется деление проектов на группы. Именно изменение группировки и сообщают нам в СМИ. Внутри же компании продолжают работать, а проекты ведутся непрерывно.

IT project management (РМ) — это дисциплина, что объединяет процедуры, принципы и политику ведения бизнеса. Она руководит проектом от разработки концепции до завершения проекта.

Общий (функциональный) и проектный менеджмент отличается тем, что функциональный менеджмент стабилен. Его цель: поддержать и приумножить качество и эффективность программного продукта. Есть отработанный шаблон, он работает постоянно. Проектный менеджмент изменчив. Цель проектного менеджмента: достигнуть результата любой ценой, учитывая, что есть deadline.

Причиной перехода от общего менеджмента к РМ чаще других становится надежность — вместо относительной абстрактной перспективы к предсказуемым результатам.

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины «Проектный менеджмент в решении инженерных задач» является формирование набора универсальных компетенций будущего бакалавра по направлению подготовки 08.03.01 Строительство .

Задачи освоения дисциплины: изучение методологии проектного менеджмента, освоение методов и инструментов проектного менеджмента в решении инженерных задач.

2. ОБОРУДОВАНИЕ И МАТЕРИАЛЫ

Аппаратные средства: персональный компьютер;

Программные средства: ОС MS Windows; MS Visual Studio, MS Office.

Учебный класс оснащен IBM-совместимыми компьютерами, объединенными в локальную сеть. Локальная сеть учебного класса имеет постоянный доступ к сети Internet по выделенной линии. Для проведения лабораторных работ необходимо следующее программное обеспечение: операционная система MS Windows, пакет офисных программ MS Office, пакет MS Visual Studio.

3. Наименование практических занятий

№ Темы дисциплины	Наименование тем дисциплины, их краткое содержание	Объем часов	Из них практическая подготовка, часов
	Раздел 1. Основы проектного менеджмента		
1	Тема 1. Введение в проектный менеджмент Методы и средства проектного менеджмента инженерных задач. Технологии проектного менеджмента инженерных задач. Функциональные и технологические требования к инженерным проектам.	3	
	Итого	3	

СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

Практическая работа 1. Введение в проектный менеджмент

Цель работы: составить и проанализировать требования к инженерному проекту.

Основы теории

Проблемы, которые приходится решать специалистам в процессе создания программного обеспечения, очень сложны. Природа этих проблем не всегда ясна, особенно если разрабатываемая программная система инновационная. В частности, трудно чётко описать те действия, которые должна выполнять система. Описание функциональных возможностей и ограничений, накладываемых на систему, называется требованиями к этой системе, а сам процесс формирования, анализа, документирования и проверки этих функциональных возможностей и ограничений – разработкой требований.

Требования подразделяются на пользовательские и системные. Пользовательские требования – это описание на естественном языке (плюс поясняющие диаграммы) функций, выполняемых системой, и ограничений, накладываемых на неё. Системные требования – это описание особенностей системы (архитектура системы, требования к параметрам оборудования и т.д.), необходимых для эффективной реализации требований пользователя.

Разработка требований

Разработка требований — это процесс, включающий мероприятия, необходимые для создания и утверждения документа, содержащего спецификацию системных требований. Различают четыре основных этапа процесса разработки требований:

- анализ технической осуществимости создания системы,
- формирование и анализ требований,
- специфицирование требований и создание соответствующей документации,
- аттестация этих требований.

На рисунке 1.1 показаны взаимосвязи между этими этапами и результаты, сопровождающие каждый этап процесса разработки системных требований.

Но поскольку в процессе разработки системы в силу разнообразных причин требования могут меняться, управление требованиями, т.е. процесс управления изменениями системных требований, является необходимой составной частью деятельности по их разработке.

Формирование и анализ требований

Следующим этапом процесса разработки требований является формирование (определение) и анализ требований.

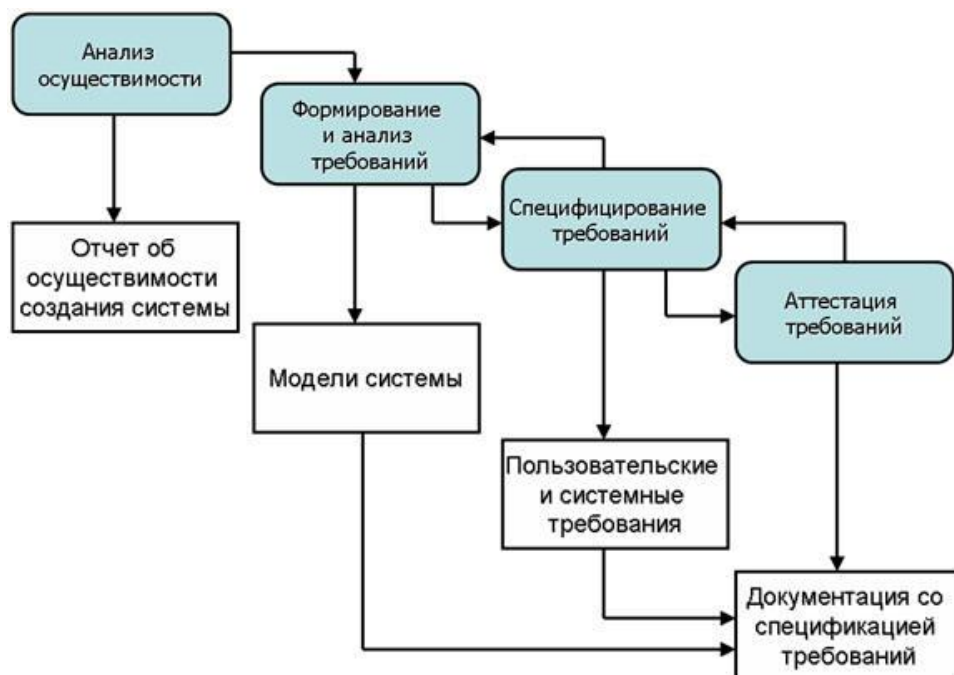


Рисунок 1.1 - Процесс разработки требований

Обобщенная модель процесса формирования и анализа требований показана на рисунке 1.2. Каждая организация использует собственный вариант этой модели, зависящий от “местных факторов”: опыта работы коллектива разработчиков, типа разрабатываемой системы, используемых стандартов и т.д.

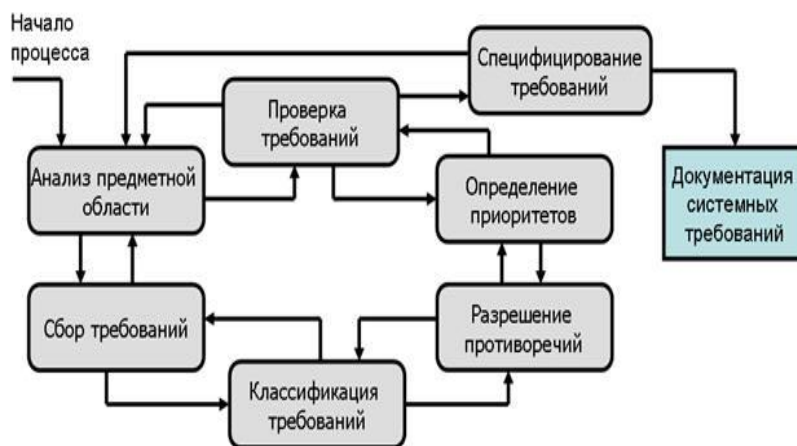


Рисунок 1.2 - Процесс формирования и анализа требований

Процесс формирования и анализа требований проходит через ряд этапов.

1. Анализ предметной области. Аналитики должны изучить предметную область, где будет эксплуатироваться система.

2. Сбор требований. Это процесс взаимодействия с лицами, формирующими требования. Во время этого процесса продолжается анализ предметной области.

3. Классификация требований. На этом этапе бесформенный набор требований преобразуется в логически связанные группы требований.

4. Разрешение противоречий. Без сомнения, требования многочисленных лиц, занятых в процессе формирования требований, будут противоречивыми. На этом этапе определяются и разрешаются противоречия различного рода.

5. Назначение приоритетов. В любом наборе требований одни из них будут более важны, чем другие. На этом этапе совместно с лицами, формирующими требования, определяются наиболее важные требования.

6. Проверка требований. На этом этапе определяется их полнота, последовательность и непротиворечивость.

Процесс формирования и анализа требований циклический, с обратной связью от одного этапа к другому. Цикл начинается с анализа предметной области и заканчивается проверкой требований. Понимание требований предметной области увеличивается в каждом цикле процесса формирования требований.

Рассмотрим три основных подхода к формированию требований: метод, основанный на множестве опорных точек зрения, сценарии и этнографический метод.

Опорные точки зрения

Подход с использованием различных опорных точек зрения к разработке требований признает различные (опорные) точки зрения на проблему и использует их в качестве основы построения и организации как процесса формирования требований, так и непосредственно самих требований.

Различные методы предлагают разные трактовки выражения "точка зрения". Точки зрения можно трактовать следующим образом.

1. Как источник информации о системных данных. В этом случае на основе опорных точек зрения строится модель создания и использования данных в системе. В процессе формирования требований отбираются все такие точки зрения (и на их основе определяются данные), которые будут созданы или использованы при работе системы, а также способы обработки этих данных.

2. Как структура представлений. В этом случае точки зрения рассматриваются как особая часть модели системы. Например, на основе различных точек зрения могут разрабатываться модели "сущность-связь", модели конечного автомата и т.д.

3. Как получатели системных сервисов. В этом случае точки зрения являются внешними (относительно системы) получателями системных сервисов. Точки зрения помогают определить данные, необходимые для выполнения системных сервисов или их управления.

Наиболее эффективным подходом к анализу таких систем является использование внешних опорных точек зрения. На основе этого подхода разработан метод VORD (Viewpoint-Oriented Requirements Definition — определение требований на основе точек зрения) для формирования и анализа требований. Основные этапы метода VORD показаны на рисунке 1.3.

1. Идентификация точек зрения, получающих системные сервисы, и идентификация сервисов, соответствующих каждой точке зрения.

2. Структурирование точек зрения — создание иерархии сгруппированных точек зрения. Общесистемные сервисы предоставляются более высоким уровням иерархии и наследуются точками зрения низшего уровня.

3. Документирование опорных точек зрения, которое заключается в точном описании идентифицированных точек зрения и сервисов.

4. Отображение системы точек зрения, которая показывает системные объекты, определенные на основе информации, заключенной в опорных точках зрения.



Рисунок 1.3 - Метод VORD

Пример. Рассмотрим использование метода VORD на первых трех шагах анализа требований для системы поддержки заказа и учета товаров в бакалейной лавке. В бакалейной лавке для каждого товара фиксируется место хранения (определенная полка), количество товара и его поставщик. Система поддержки заказа и учета товаров должна обеспечивать добавление информации о новом товаре, изменение или удаление информации об имеющемся товаре, хранение (добавление, изменение и удаление) информации о поставщиках, включающей в себя название фирмы, ее адрес и телефон. При помощи системы составляются заказы поставщикам. Каждый заказ может содержать несколько позиций, в каждой позиции указываются наименование товара и его количество в заказе. Система по требованию пользователя формирует и выдает на печать следующую справочную информацию:

- список всех товаров;
- список товаров, имеющихся в наличии;
- список товаров, количество которых необходимо пополнить;
- список товаров, поставляемых данным поставщиком.

Первым шагом в формировании требований является идентификация опорных точек зрения. Во всех методах формирования требований, основанных на использовании точек зрения, начальная идентификация является наиболее трудной задачей. Один из подходов к идентификации точек зрения — метод "мозговой атаки", когда определяются потенциальные системные сервисы и организации, взаимодействующие с системой. Организуется встреча лиц, участвующих в формировании требований, которые предлагают свои точки зрения. Эти точки зрения представляются в виде диаграммы, состоящей из ряда круговых областей, отображающих возможные точки зрения (рис. 4). Во время "мозговой атаки" необходимо идентифицировать потенциальные опорные точки зрения, системные сервисы, входные данные, нефункциональные требования, управляющие события и исключительные ситуации.

Следующей стадией процесса формирования требований будет идентификация опорных точек зрения (на рисунке 1.4 показаны в виде темных круговых областей) и сервисов (показаны в виде затененных областей). Сервисы должны соответствовать опорным точкам зрения. Но могут быть сервисы, которые не поставлены им в соответствие. Это означает, что на начальном этапе "мозговой атаки" некоторые опорные точки зрения не были идентифицированы.

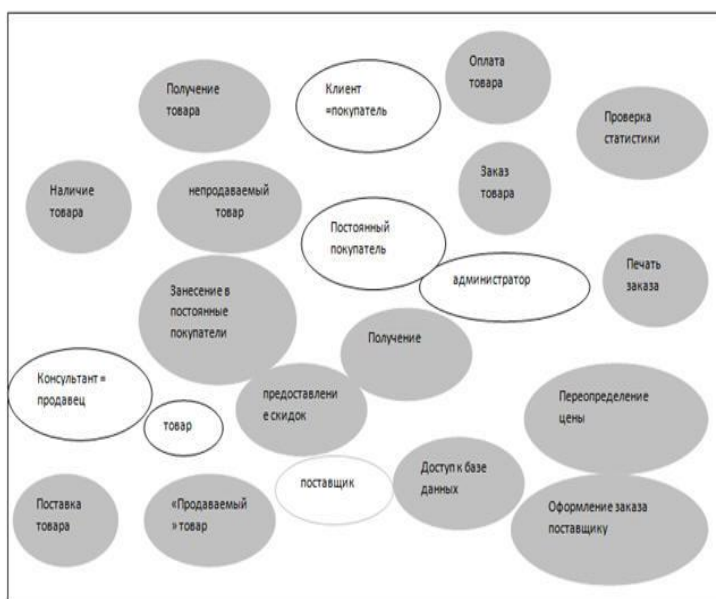


Рисунок 1.4 - Диаграмма идентификации точек зрения

В таблице 1.1 показано распределение сервисов для некоторых идентифицированных на рисунке 1.4 точек зрения. Один и тот же сервис может быть соотнесен с несколькими точками зрения.

Таблица 1.1 - Сервисы, соотнесенные с точками зрения

клиент	покупатель	постоянный покупатель	товар	поставщик	продавец	администратор
Проверка наличия товара	Занесение в список клиентов	Получение скидки	Прием товара	Занесение в базу данных	Продажа товара	Доступ к базе данных
Покупка товара		Получение информации	Занесение в базу данных		Печать чека	Проверка статистики
Получение чека			Назначение цены		Доступ к каталогу	Переопределение цены
Заказ товара			Переопределение цены		Проверка наличия товара	Оформление заказа
Занесение покупателя и суммы покупки в базу данных			«Покупаемый» или «непокупаемый» товар		Оформление заказа покупателю	Печать заказа

Информация, извлеченная из точек зрения, используется для заполнения форм шаблонов точек зрения и организации точек зрения в иерархию наследования. Это позволяет увидеть общие точки зрения и повторно использовать информацию в иерархии наследования. Сервисы, данные и управляющая информация наследуются подмножеством точек зрения. На рисунке 1.5 показана часть иерархии точек зрения для системы поддержки заказа и учета товаров.

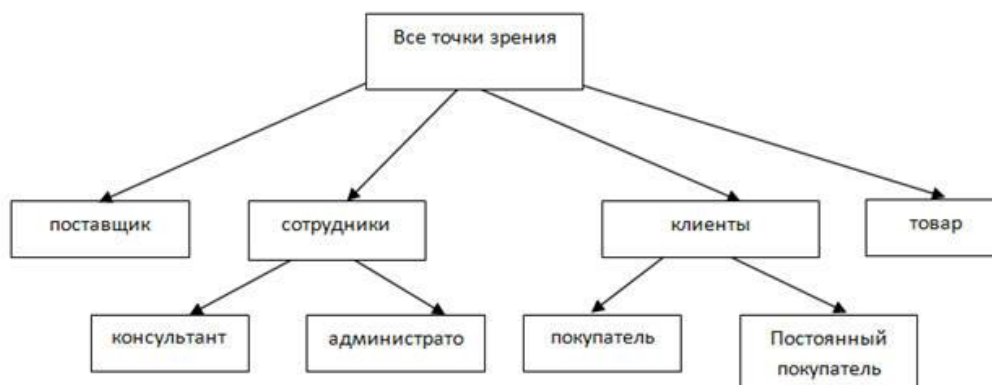


Рисунок 1.5 - Иерархия точек зрения

Аттестация требований

Аттестация должна продемонстрировать, что требования действительно определяют ту систему, которую хочет иметь заказчик. Проверка требований важна, так

как ошибки в спецификации требований могут привести к переделке системы и большим затратам, если будут обнаружены во время процесса разработки системы или после введения ее в эксплуатацию. Стоимость внесения в систему изменений, необходимых для устранения ошибок в требованиях, намного выше, чем исправление ошибок проектирования или кодирования. Причина в том, что изменение требований обычно влечет за собой значительные изменения в системе, после внесения которых она должна пройти повторное тестирование.

Во время процесса аттестации должны быть выполнены различные типы проверок требований.

1. Проверка правильности требований. Пользователь может считать, что система необходима для выполнения некоторых определенных функций. Однако дальнейшие размышления и анализ могут привести к необходимости введения дополнительных или новых функций. Системы предназначены для разных пользователей с различными потребностями, и поэтому набор требований будет представлять собой некоторый компромисс между требованиями пользователей системы.

2. Проверка на непротиворечивость. Спецификация требований не должна содержать противоречий. Это означает, что в требованиях не должно быть противоречащих друг другу ограничений или различных описаний одной и той же системной функции.

3. Проверка на полноту. Спецификация требований должна содержать требования, которые определяют все системные функции и ограничения, налагаемые на систему.

4. Проверка на выполнимость. На основе знания существующих технологий требования должны быть проверены на возможность их реального выполнения. Здесь также проверяются возможности финансирования и график разработки системы.

Существует ряд методов аттестации требований, которые можно использовать совместно или каждый в отдельности.

1. Обзор требований. Требования системно анализируются рецензентами.

2. Прототипирование. На этом этапе прототип системы демонстрируется конечным пользователям и заказчику. Они могут экспериментировать с этим прототипом, чтобы убедиться, что он отвечает их потребностям.

3. Генерация тестовых сценариев. В идеале требования должны быть такими, чтобы их реализацию можно было протестировать. Если тесты для требований разрабатываются как часть процесса аттестации, то часто это позволяет обнаружить проблемы в спецификации. Если такие тесты сложно или невозможно разработать, то обычно это означает, что требования трудно выполнить и поэтому необходимо их пересмотреть.

4. Автоматизированный анализ непротиворечивости. Если требования представлены в виде структурных или формальных системных моделей, можно использовать инструментальные CASE-средства для проверки непротиворечивости моделей. Для автоматизированной проверки непротиворечивости необходимо построить базу данных требований и затем проверить все требования в этой базе данных. Анализатор требований готовит отчет обо всех обнаруженных противоречиях.

Пользовательские и системные требования

На основании полученных моделей строятся пользовательские требования, т.е. как было сказано в начале описание на естественном языке функции, выполняемых системой, и ограничений, накладываемых на неё.

Пользовательские требования должны описывать внешнее поведение системы, основные функции и сервисы предоставляемые системой, её нефункциональные свойства. Необходимо выделить опорные точки зрения и сгруппировать требования в соответствии с ними. Пользовательские требования можно оформить как простым перечислением, так и используя нотацию вариантов использования.

Далее составляются системные требования. Они включают в себя:

1. Требования к архитектуре системы. Например, число и размещение хранилищ и серверов приложений.

2. Требования к параметрам оборудования. Например, частота процессоров серверов и клиентов, объём хранилищ, размер оперативной и видео памяти, пропускная способность канала и т.д.

3. Требования к параметрам системы. Например, время отклика на действие пользователя, максимальный размер передаваемого файла, максимальная скорость передачи данных, максимальное число одновременно работающих пользователей и т.д.

4. Требования к программному интерфейсу.

5. Требования к структуре системы. Например, Масштабируемость, распределённость, модульность, открытость.

- масштабируемость – возможность распространения системы на большое количество машин, не приводящая к потере работоспособности и эффективности, при этом способность системы наращивать свою мощность должна определяться только мощностью соответствующего аппаратного обеспечения.

- распределенность - система должна поддерживать распределённое хранение данных.

- модульность - система должна состоять из отдельных модулей, интегрированных между собой.

- открытость - наличие открытых интерфейсов для возможной доработки и интеграции с другими системами.

6. Требования по взаимодействию и интеграции с другими системами. Например, использование общей базы данных, возможность получения данных из баз данных определённых систем и т.д.

Постановка задачи к практической работе 1

1. Изучить предлагаемый теоретический материал.

2. Построить опорные точки зрения на основании метода VORD для формирования и анализа требований. Результатом должны явиться две диаграммы: диаграмма идентификации точек зрения и диаграмма иерархии точек зрения.

3. Составить информационную модель будущей системы, включающую в себя описание основных объектов системы и взаимодействия между ними. На основании информационной модели, диаграммы идентификации точек зрения, диаграммы иерархии точек зрения сформировать требования пользователя и системные требования.

4. Оформить отчет. Представить отчет для защиты.

Варианты индивидуальных заданий

В соответствии с указанной предметной областью и классом разрабатываемой системы разработать задание на инженерный проект.

Таблица 1.2 – Индивидуальные задания

№	Предметная область	Класс ИС
1	Склад	MRP
2	Производственное предприятие	ERP
3	Торговое предприятие	CRM
4	Торговое предприятие	SCM
5	Торговое предприятие	B2C
6	Портал	B2B
7	Строительное предприятие	ИС учета
8	Высшее учебное заведение	ИС управления
9	Инфраструктура предприятия	СППР
10	Аппаратная инфраструктура предприятия	Экспертная система

Содержание отчета

По выполненной работе составляется отчет. Отчет выполняется в электронном виде. По выполненному отчету проводится защита практической работы.

Отчет по практической работе должен состоять из следующих структурных элементов:

- титульный лист;
- вводная часть;
- основная часть;
- заключение (выводы).

Вводная часть отчета должна включать пункты:

- условие задачи;
- порядок выполнения.
- программно-аппаратные средства, используемые при выполнении работы.

Защита отчета по заключается в предьявлении преподавателю полученных результатов в виде файла и демонстрации полученных навыков при ответах на вопросы преподавателя.

Контрольные вопросы

1. Что такое жизненный цикл инженерного проекта?
2. Дайте определение модели жизненного цикла инженерного проекта.
3. Приведите этапы разработки инженерного проекта.
4. Какие этапы включает в себя модель ЖЦ ПО согласно ГОСТ 19.102-77?
5. Что включает в себя этап предпроектного исследования?
6. Перечислите функциональные требования к инженерному проекту.
7. Перечислите эксплуатационные требования к инженерному проекту.
8. Перечислите правила разработки технического задания.
9. Назовите основные разделы технического задания.
10. Назовите этапы выработки требований к инженерному проекту.

Практическая работа 2. Методы и средства проектного менеджмента

Цель работы: изучение методов и средств проектного менеджмента инженерных решений; применение методов и средств проектного менеджмента.

Практическая работа направлена на ознакомление с процессом описания инженерного проекта и получение навыков по использованию основных методов анализа инженерного решения.

Требования к результатам выполнения практической работы:

- наличие описания инженерного проекта;
- проведение анализа осуществимости выполнения проекта;
- наличие заключения о возможности реализации проекта, содержащего рекомендации относительно разработки системы, базовые предложения по объёму требуемого бюджета, числу разработчиков, времени и требуемому программному обеспечению.

Основы теории

Проблемы управления инженерными проектами впервые проявились в 60-х - начале 70-х годов, когда провалились многие большие проекты по разработке программных продуктов. Были зафиксированы задержки в создании ПО, оно было ненадежным, затраты на разработку в несколько раз превосходили первоначальные оценки, созданные программные системы часто имели низкие показатели производительности. Причины провалов коренились в тех подходах, которые использовались в управлении проектами. Применяемая методика была основана на опыте

управления техническими проектами и оказалась неэффективной при разработке программного обеспечения.

Важно понимать разницу между профессиональной разработкой ПО и любительским программированием. Необходимость управления программными проектами вытекает из того факта, что процесс создания профессионального ПО всегда является субъектом бюджетной политики организации, где оно разрабатывается, и имеет временные ограничения. Работа руководителя программного проекта по большому счету заключается в том, чтобы гарантировать выполнение этих бюджетных и временных ограничений с учетом бизнес-целей организации относительно разрабатываемого ПО.

Руководители проектов призваны спланировать все этапы разработки программного продукта. Они также должны контролировать ход выполнения работ и соблюдения всех требуемых стандартов. Постоянный контроль за ходом выполнения работ необходим для того, чтобы процесс разработки не выходил за временные и бюджетные ограничения. Хорошее управление не гарантирует успешного завершения проекта, но плохое управление обязательно приведет к его провалу. Это может выразиться в задержке сроков сдачи готового ПО, в превышении сметной стоимости проекта и в несоответствии готового ПО спецификации требований.

Процесс разработки ПО существенно отличается от процессов реализации технических проектов, что порождает определенные сложности в управлении программными проектами:

1. Программный продукт нематериален. Программное обеспечение нематериально, его нельзя увидеть или потрогать. Руководитель программного проекта не видит процесс "роста" разрабатываемого ПО. Он может полагаться только на документацию, которая фиксирует процесс разработки программного продукта.

2. Не существует стандартных процессов разработки ПО. На сегодняшний день не существует четкой зависимости между процессом создания ПО и типом создаваемого программного продукта. Другие технические дисциплины имеют длительную историю, процессы разработки технических изделий многократно опробованы и проверены. Процессы создания большинства технических систем хорошо изучены. Изучением же процессов создания ПО специалисты занимаются только последнее время. Поэтому пока нельзя точно предсказать, на каком этапе процесса разработки ПО могут возникнуть проблемы, угрожающие всему программному проекту.

3. Большие программные проекты - это часто "одноразовые" проекты. Большие программные проекты, как правило, значительно отличаются от проектов, реализованных ранее. Поэтому, чтобы уменьшить неопределенность в планировании проекта, руководители проектов должны обладать очень большим практическим опытом. Но постоянные технологические изменения в компьютерной технике и коммуникационном оборудовании обесценивают предыдущий опыт. Знания и навыки, накопленные опытом, могут не востребоваться в новом проекте.

Перечисленные отличия могут привести к тому, что реализация проекта выйдет из временного графика или превысит бюджетные ассигнования. Программные системы зачастую оказываются новинками как в "идеологическом", так и в техническом плане. Поэтому, предвидя возможные проблемы в реализации программного проекта, следует всегда помнить, что многим из них свойственно выходить за рамки временных и бюджетных ограничений.

Процесс управления разработкой инженерного проекта

Невозможно описать и стандартизировать все работы, выполняемые в проекте по созданию ПО. Эти работы весьма существенно зависят от организации, где выполняется разработка ПО, и от типа создаваемого программного продукта. Но всегда можно выделить следующие:

- Написание предложений по созданию ПО.
- Планирование и составление графика работ по созданию ПО.

- Оценивание стоимости проекта.
- Подбор персонала.
- Контроль за ходом выполнения работ.
- Написание отчетов и представлений.

Первая стадия программного проекта может состоять из написания предложений по реализации этого проекта. Предложения должны содержать описание целей проектов и способов их достижения. Они также обычно включают в себя оценки финансовых и временных затрат на выполнение проекта. При необходимости здесь могут приводиться обоснования для передачи проекта на выполнение сторонней организации или команде разработчиков.

Написание предложений — очень ответственная работа, так как для многих организаций вопрос о том, будет ли проект выполняться самой организацией или разрабатываться по контракту сторонней компанией, является критическим. Не существует каких-либо рекомендаций по написанию предложений, многое здесь зависит от опыта.

На этапе планирования проекта определяются процессы, этапы и полученные на каждом из них результаты, которые должны привести к выполнению проекта. Реализация этого плана приведет к достижению целей проекта. Определение стоимости проекта напрямую связано с его планированием, поскольку здесь оцениваются ресурсы, требующиеся для выполнения плана.

Контроль за ходом выполнения работ (мониторинг проекта) — это непрерывный процесс, продолжающийся в течение всего срока реализации проекта. Руководитель должен постоянно отслеживать ход реализации проекта и сравнивать фактические и плановые показатели выполнения работ с их стоимостью. Хотя многие организации имеют механизмы формального мониторинга работ, опытный руководитель может составить ясную картину о стадии развития проекта просто путем неформального общения с разработчиками.

Неформальный мониторинг часто помогает обнаружить потенциальные проблемы, которые в явном виде могут обнаружиться позднее. Например, ежедневное обсуждение хода выполнения работ может выявить отдельные недоработки в создаваемом программном продукте. Вместо ожидания отчетов, в которых будет отражен факт "пробуксовки" графика работ, можно обсудить со специалистами намечающиеся программистские проблемы и не допустить срыва графика работ.

В течение реализации проекта обычно происходит несколько формальных контрольных проверок хода выполнения работ по созданию ПО. Такие проверки должны дать общую картину хода реализации проекта в целом и показать, насколько уже разработанная часть ПО соответствует целям проекта.

Время выполнения больших программных проектов может занимать несколько лет. В течение этого времени цели и намерения организации, заказавшей программный проект, могут существенно измениться. Может оказаться, что разрабатываемый программный продукт стал уже ненужным либо исходные требования к создаваемому ПО просто устарели и их необходимо кардинально менять. В такой ситуации руководство организации-разработчика может принять решение о прекращении разработки ПО или об изменении проекта в целом с тем, чтобы учесть изменившиеся цели и намерения организации-заказчика.

Руководители проектов обычно обязаны сами подбирать исполнителей для своих проектов. В идеальном случае профессиональный уровень исполнителей должен соответствовать той работе, которую они будут выполнять в ходе реализации проекта. Однако во многих случаях руководители должны полагаться на команду разработчиков, которая далека от идеальной. Такая ситуация может быть вызвана следующими причинами:

1. Бюджет проекта не позволяет привлечь высококвалифицированный персонал. В таком случае за меньшую плату привлекаются менее квалифицированные специалисты.

2. Бывают ситуации, когда невозможно найти специалистов необходимой квалификации как в самой организации-разработчике, так и вне ее. Например, в организации "лучшие люди" могут быть уже заняты в других проектах.

3. Организация хочет повысить профессиональный уровень своих работников. В этом случае она может привлечь к участию в проекте неопытных или недостаточно квалифицированных работников, чтобы они приобрели необходимый опыт и поучились у более опытных специалистов.

Таким образом, почти всегда подбор специалистов для выполнения проекта имеет определенные ограничения и не является свободным. Вместе с тем необходимо, чтобы хотя бы несколько членов группы разработчиков имели квалификацию и опыт, достаточные для работы над данным проектом. В противном случае невозможно избежать ошибок в разработке ПО.

Руководитель проекта обычно обязан посылать отчеты о ходе его выполнения как заказчику, так и подрядным организациям. Это должны быть краткие документы, основанные на информации, извлекаемой из подробных отчетов о проекте. В этих отчетах должна быть та информация, которая позволяет четко оценить степень готовности создаваемого программного продукта.

В рамках курса «Системная инженерия» выделены следующие роли в группе по разработке ПО:

- Руководитель – общее руководство проектом, написание документации, общение с заказчиком ПО;
- Системный аналитик – разработка требований (составление технического задания, проекта программного обеспечения);
- Тестер – составление плана тестирования и аттестации готового ПО (продукта), составление сценария тестирования, базовый пример, проведение мероприятий по плану тестирования;
- Разработчик – моделирование компонент программного обеспечения, кодирование.

Планирование проекта разработки инженерного проекта

Эффективное управление программным проектом напрямую зависит от правильного планирования работ, необходимых для его выполнения. План помогает руководителю предвидеть проблемы, которые могут возникнуть на каких-либо этапах создания ПО, и разработать превентивные меры для их предупреждения или решения. План, разработанный на начальном этапе проекта, рассматривается всеми его участниками как руководящий документ, выполнение которого должно привести к успешному завершению проекта. Этот первоначальный план должен максимально подробно описывать все этапы реализации проекта.

Процесс планирования начинается, исходя из описания системы, с определения проектных ограничений (временные ограничения, возможности наличного персонала, бюджетные ограничения и т.д.). Эти ограничения должны определяться параллельно с оцениванием проектных параметров, таких как структура и размер проекта, а также распределением функций среди исполнителей. Затем определяются этапы разработки и то, какие результаты документация, прототипы, подсистемы или версии программного продукта) должны быть получены по окончании этих этапов. Далее начинается циклическая часть планирования. Сначала разрабатывается график работ по выполнению проекта или дается разрешение на продолжение использования ранее созданного графика. После этого проводится контроль выполнения работ и отмечаются расхождения между реальным и плановым ходом работ.

Далее, по мере поступления новой информации о ходе выполнения проекта, возможен пересмотр первоначальных оценок параметров проекта. Это, в свою очередь,

может привести к изменению графика работ. Если в результате этих изменений нарушаются сроки завершения проекта, должны быть пересмотрены (и согласованы с заказчиком ПО) проектные ограничения.

Конечно, большинство руководителей проектов не думают, что реализация их проектов пройдет гладко, без всяких проблем. Желательно описать возможные проблемы еще до того, как они проявят себя в ходе выполнения проекта. Поэтому лучше составлять "пессимистические" графики работ, чем "оптимистические". Но, конечно, невозможно построить план, учитывающий все, в том числе случайные, проблемы и задержки выполнения проекта, поэтому и возникает необходимость периодического пересмотра проектных ограничений и этапов создания программного продукта.

План проекта должен четко показать ресурсы, необходимые для реализации проекта, разделение работ на этапы и временной график выполнения этих этапов. В некоторых организациях план проекта составляется как единый документ, содержащий все виды планов, описанных выше. В других случаях план проекта описывает только технологический процесс создания ПО. В таком плане обязательно присутствуют ссылки на планы других видов, но они разрабатываются отдельно от плана проекта.

Детализация планов проектов очень разнится в зависимости от типа разрабатываемого программного продукта и организации-разработчика. Но в любом случае большинство планов содержат следующие разделы.

1. Введение. Краткое описание целей проекта и проектных ограничений (бюджетных, временных и т.д.), которые важны для управления проектом.

2. Организация выполнения проекта. Описание способа подбора команды разработчиков и распределение обязанностей между членами команды.

3. Анализ рисков. Описание возможных проектных рисков, вероятности их проявления и стратегий, направленных на их уменьшение.

4. Аппаратные и программные ресурсы, необходимые для реализации проекта. Перечень аппаратных средств и программного обеспечения, необходимого для разработки программного продукта. Если аппаратные средства требуется закупать, приводится их стоимость совместно с графиком закупки и поставки.

5. Разбиение работ на этапы. Процесс реализации проекта разбивается на отдельные процессы, определяются этапы выполнения проекта, приводится описание результатов ("выходов") каждого этапа и контрольные отметки.

6. График работ. В этом графике отображаются зависимости между отдельными процессами (этапами) разработки ПО, оценки времени их выполнения и распределение членов команды разработчиков по отдельным этапам.

7. Механизмы мониторинга и контроля за ходом выполнения проекта. Описываются предоставляемые руководителем отчеты о ходе выполнения работ, сроки их предоставления, а также механизмы мониторинга всего проекта.

План должен регулярно пересматриваться в процессе реализации проекта. Одни части плана, например график работ, изменяются часто, другие более стабильны. Для внесения изменений в план требуется специальная организация документопотока, позволяющая отслеживать эти изменения.

УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Перечень основной и дополнительной литературы, необходимой для освоения дисциплины

Перечень основной литературы

1. Анисимов, А.А. Менеджмент в сфере информационной безопасности : курс лекций / А.А. Анисимов. - М. : Интернет-Университет Информационных Технологий,

2009. - 176 с. - (Основы информационных технологий). - ISBN 9778-5-9963-0237-6 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=232981>.

2. Информационный менеджмент : учебное пособие / под ред. Е.Н. Барикаев, Г.Г. Чараев. - М. : Юнити-Дана, 2012. - 360 с. - ISBN 978-5-238-02328-1 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=119528>.

Перечень дополнительной литературы

1 Кулешов, А. В. Контракты и внешнеторговая документация : учеб. пособие / А.В. Кулешов, Л.А. Желтова, О.В. Шишкина. - СПб. : Троицкий мост, 2012. - 256 с. : ил. - На учебнике гриф: Доп.УМО. - Прил.: с. 211-256. - Библиогр.: с. 208-210. - ISBN 978-5-4377-0002-0

2 Арустамов, Э. А. Внешнеэкономическая деятельность : учебник / Э.А. Арустамов, Р.С. Андреева. - М. : КНОРУС, 2011. - 168 с. - (Среднее профессиональное образование). - Прил.: с. 158-165. - Библиогр.: с. 146-147. - ISBN 978-5-406-01065-5

Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине

1. Методические указания по выполнению практических работ по дисциплине «Проектный менеджмент в решении инженерных задач»

2. Методические рекомендации для студентов по организации самостоятельной работы по дисциплине «Проектный менеджмент в решении инженерных задач»