

Управление качеством проекта (окончание)

Лектор – Скороход С. В.

Нефункциональное тестирование

Тестирование удобства пользования - это метод тестирования, направленный на установление степени удобства использования, обучаемости, понятности и привлекательности для пользователей разрабатываемого продукта в контексте заданных условий.

Тестирование удобства пользования дает оценку уровня удобства использования приложения по следующим пунктам:

- производительность, эффективность - сколько времени и шагов понадобится пользователю для завершения основных задач приложения, например, размещение новости, регистрации, покупка и т.д.?
- правильность - сколько ошибок сделал пользователь во время работы с приложением?
- активизация в памяти — как много пользователь помнит о работе приложения после приостановки работы с ним на длительный период времени? (повторное выполнение операций после перерыва должно проходить быстрее чем у нового пользователя).
- эмоциональная реакция — как пользователь себя чувствует после завершения задачи - растерян, испытал стресс? Порекомендует ли пользователь систему своим друзьям? (положительная реакция - лучше)

Нефункциональное тестирование

Тестирование на отказ и восстановление (Failover and Recovery Testing)

проверяет тестируемый продукт с точки зрения способности противостоять и успешно восстанавливаться после возможных сбоев, возникших в связи с ошибками программного обеспечения, отказами оборудования или проблемами связи.

Тестирование на отказ и восстановление очень важно для систем, работающих по принципу “24x7”. Если Вы создаете продукт, который будет работать, например в интернете, то без проведения данного вида тестирования просто не обойтись. Т.к. каждая минута простоя или потеря данных в случае отказа оборудования, может стоить денег, потери клиентов и репутации на рынке.

Методика подобного **тестирования** заключается в симулировании различных условий сбоя и последующем изучении и оценке реакции защитных систем. В процессе подобных проверок выясняется, была ли достигнута требуемая степень восстановления системы после возникновения сбоя.

Нефункциональное тестирование

Конфигурационное тестирование — специальный вид тестирования, направленный на проверку работы программного обеспечения при различных конфигурациях системы (заявленных платформах, поддерживаемых драйверах, при различных конфигурациях компьютеров и т.д.)

В зависимости от типа проекта конфигурационное тестирование может иметь разные цели:

Проект по профилированию работы системы

- Цель Тестирования: определить оптимальную конфигурацию оборудования, обеспечивающую требуемые характеристики производительности и времени реакции тестируемой системы.

Проект по миграции системы с одной платформы на другую

- Цель Тестирования: Проверить объект тестирования на совместимость с объявленным в спецификации оборудованием, операционными системами и программными продуктами третьих фирм.

Тестирование связанное с изменениями

Понятие **дымовое тестирование** пошло из инженерной среды: «При вводе в эксплуатацию нового оборудования ("железа") считалось, что тестирование прошло удачно, если из установки не пошел дым».

В области программного обеспечения, дымовое тестирование рассматривается как короткий цикл тестов, выполняемый для подтверждения того, что после сборки кода (нового или исправленного) устанавливаемое приложение, стартует и выполняет основные функции.

Вывод о работоспособности основных функций делается на основании результатов поверхностного тестирования наиболее важных модулей приложения на предмет возможности выполнения требуемых задач и наличия быстронаходимых критических и блокирующих дефектов. В случае отсутствия таковых дефектов дымовое тестирование объявляется пройденным, и приложение передается для проведения полного цикла тестирования, в противном случае, дымовое тестирование объявляется проваленным, и приложение уходит на доработку.

Тестирование связанное с изменениями

- **Регрессионное тестирование** - это вид тестирования направленный на проверку изменений, сделанных в приложении или окружающей среде (починка дефекта, слияние кода, миграция на другую операционную систему, базу данных, веб сервер или сервер приложения), для подтверждения того факта, что существующая ранее функциональность работает как и прежде.
- Как правило, **для регрессионного тестирования используются тест кейсы, написанные на ранних стадиях разработки и тестирования.** Это дает гарантию того, что изменения в новой версии приложения не повредили уже существующую функциональность. Рекомендуется делать автоматизацию регрессионных тестов, для ускорения последующего процесса тестирования и обнаружения дефектов на ранних стадиях разработки программного обеспечения.

Выделяют 3 основных типа регрессионного тестирования:

- **Регрессия багов** - попытка доказать, что исправленная ошибка на самом деле не исправлена
- **Регрессия старых багов** - попытка доказать, что недавнее изменение кода или данных сломало исправление старых ошибок, т.е. старые баги стали снова воспроизводиться.
- **Регрессия побочного эффекта** - попытка доказать, что недавнее изменение кода или данных сломало другие части разрабатываемого приложения

Тестирование связанное с изменениями

Тестирование сборки – тестирование, направленное на определение соответствия, выпущенной версии, критериям качества для начала тестирования. По своим целям является аналогом Дымового Тестирования, направленного на приемку новой версии в дальнейшее тестирование или эксплуатацию. Вглубь оно может проникать дальше, в зависимости от требований к качеству выпущенной версии.

Санитарное тестирование - это узконаправленное тестирование достаточное для доказательства того, что конкретная функция работает согласно заявленным в спецификации требованиям. Используется для определения работоспособности определенной части приложения после изменений произведенных в ней или окружающей среде. Обычно выполняется вручную.

Уровни Тестирования Программного Обеспечения

Тестирование на разных уровнях производится на протяжении всего жизненного цикла разработки и сопровождения программного обеспечения. Уровень тестирования определяет то, **над чем** производятся тесты: над отдельным модулем, группой модулей или системой, в целом. Проведение тестирования на всех уровнях системы - это залог успешной реализации и сдачи проекта.

Уровни Тестирования:

- **Компонентное или Модульное тестирование** - проверяет функциональность и ищет дефекты в частях приложения.
- **Интеграционное тестирование** - предназначено для проверки связи между компонентами, а также взаимодействия с различными частями системы.
- **Системное тестирование** - проверка как функциональных, так и не функциональных требований в системе в целом.
- **Приемочное тестирование** - формальный процесс тестирования, который проверяет соответствие системы требованиям и проводится с целью:
 - определения удовлетворяет ли система приемочным критериям;
 - вынесения решения заказчиком или другим уполномоченным лицом принимается приложение или нет.

Управление командой проекта

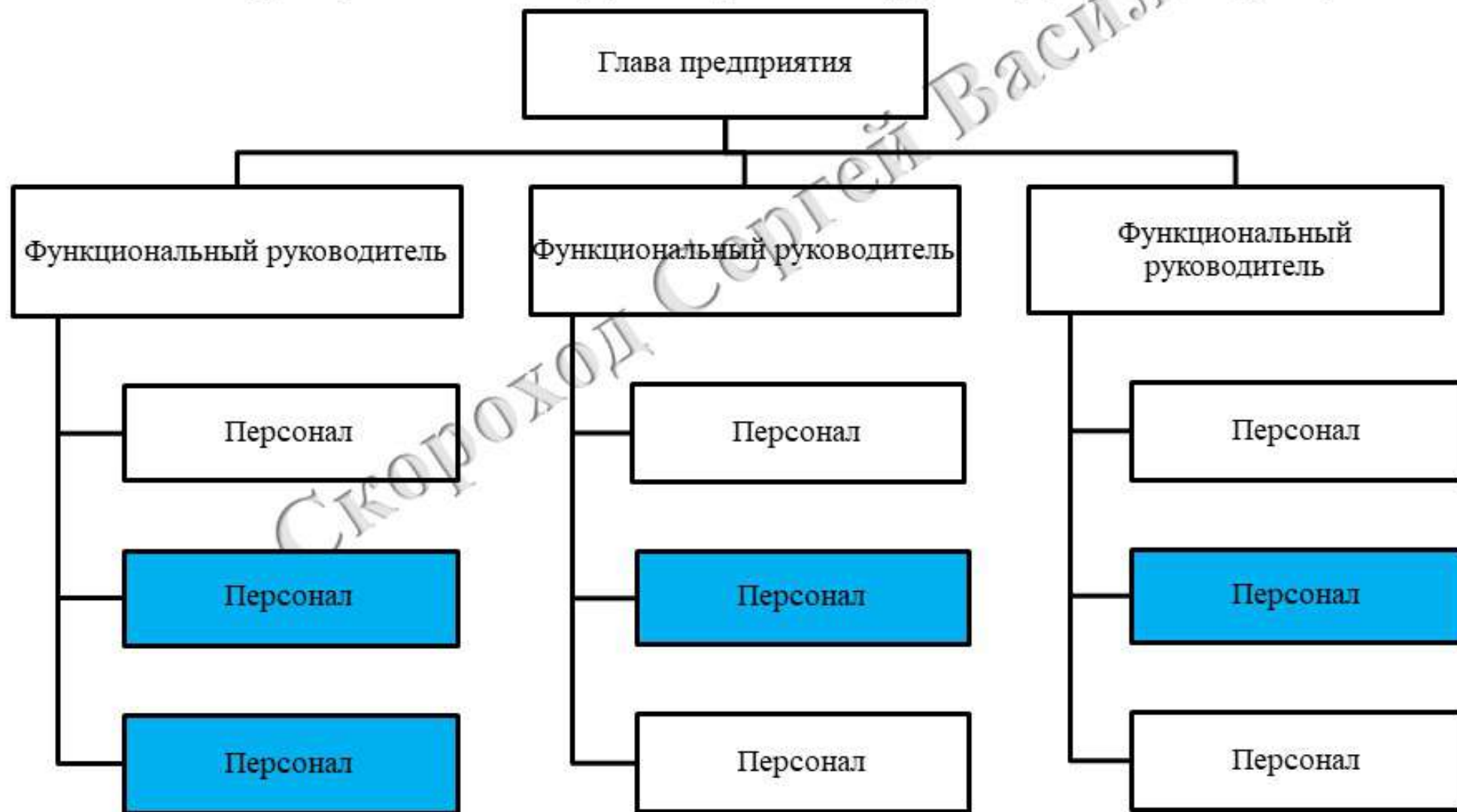
Лектор – Скороход С. В.

Организационная структура проекта

- Для выполнения проекта создается **команда проекта** - новый временный рабочий коллектив, состоящий из специалистов различных структурных подразделений компаний со стороны Исполнителя и со стороны Заказчика.
- Для членов команды проекта необходимо определить **проектные роли** (временные должности), функции, обязанности, ответственность, полномочия и правила взаимодействия, а также организационную схему, отражающую отношения подчиненности.
- **Организационная структура проекта** - это соответствующая проекту временная организационная структура, включающая всех его участников и создаваемая для успешного управления и достижения целей проекта.
- Проект - временное предприятие, по окончании которого команда проекта распускается и специалисты приступают к своим функциональным обязанностям в соответствии со штатной организационной структурой компании или переходят на следующий проект, где их функции и полномочия могут быть другими.

Функциональная организация

Иерархически выстроенная организация, в которой у каждого сотрудника есть один прямой начальник, сотрудники разделены на группы (отделы) по областям специализации. Каждая группа (отдел) управляется одним человеком - функциональным руководителем (руководителем отдела)

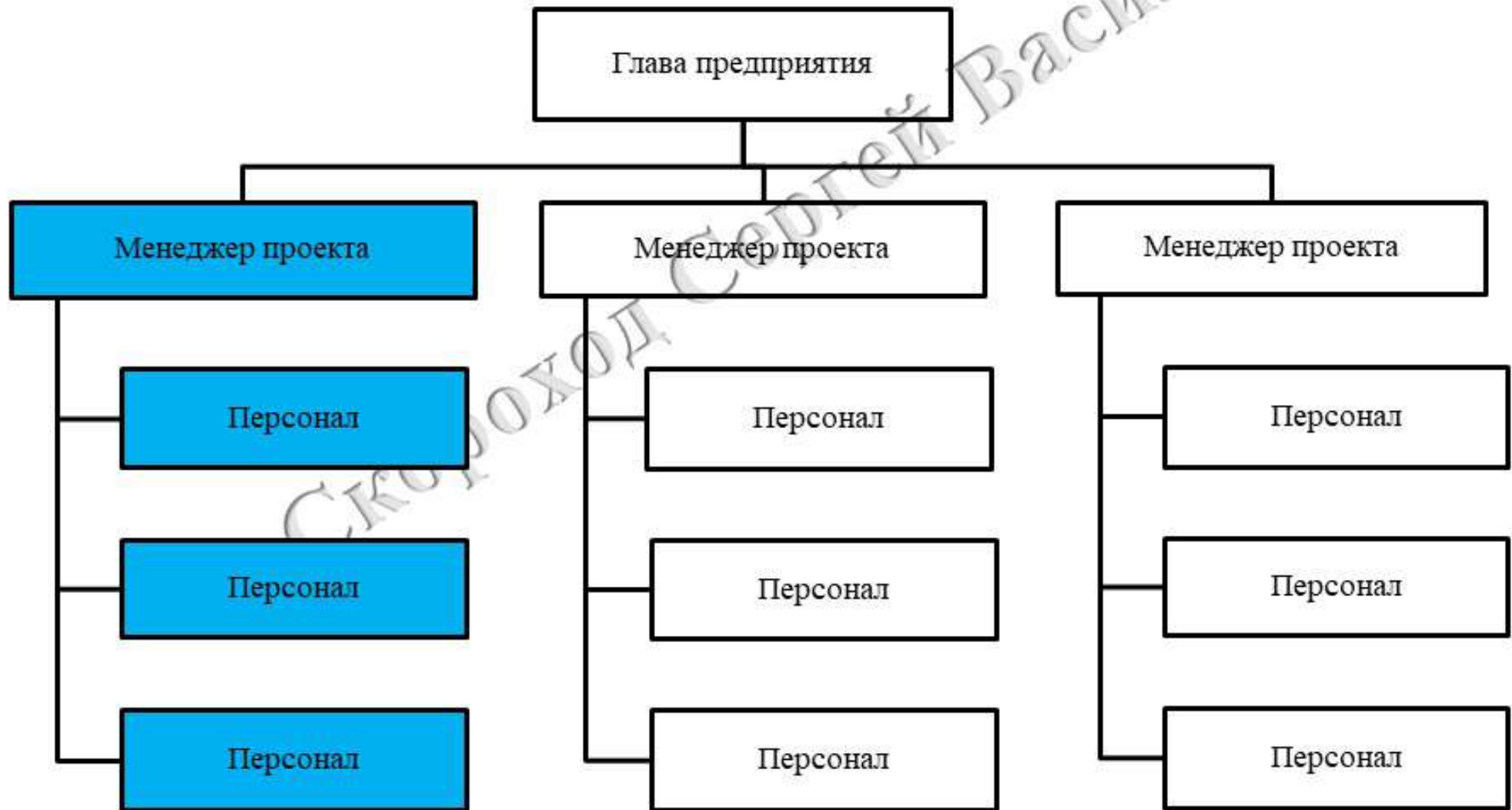


Функциональная организация

- Каждый служащий имеет одного четко выделяемого руководителя.
- Персонал группируется по специальностям, например, кодирование, маркетинг, финансы, разработка документации, тестирование.
- Когда возникает вопрос, касающийся специалистов другого отдела, служащие подают запрос своему руководителю, который консультируется с руководителем другого отдела.
- Руководитель этого отдела передает запрос своим подчиненным.
- Ответ подчиненных транслируется по обратной цепочке.
- Недостатки: нагрузка на функциональных руководителей, нет явно выраженного руководителя проекта.

Проектная организация

Руководитель проекта обладает достаточными полномочиями по установлению приоритетов, использованию ресурсов и руководству работой лиц, назначенных на исполнение проекта, а также финансовыми полномочиями в рамках бюджета проекта

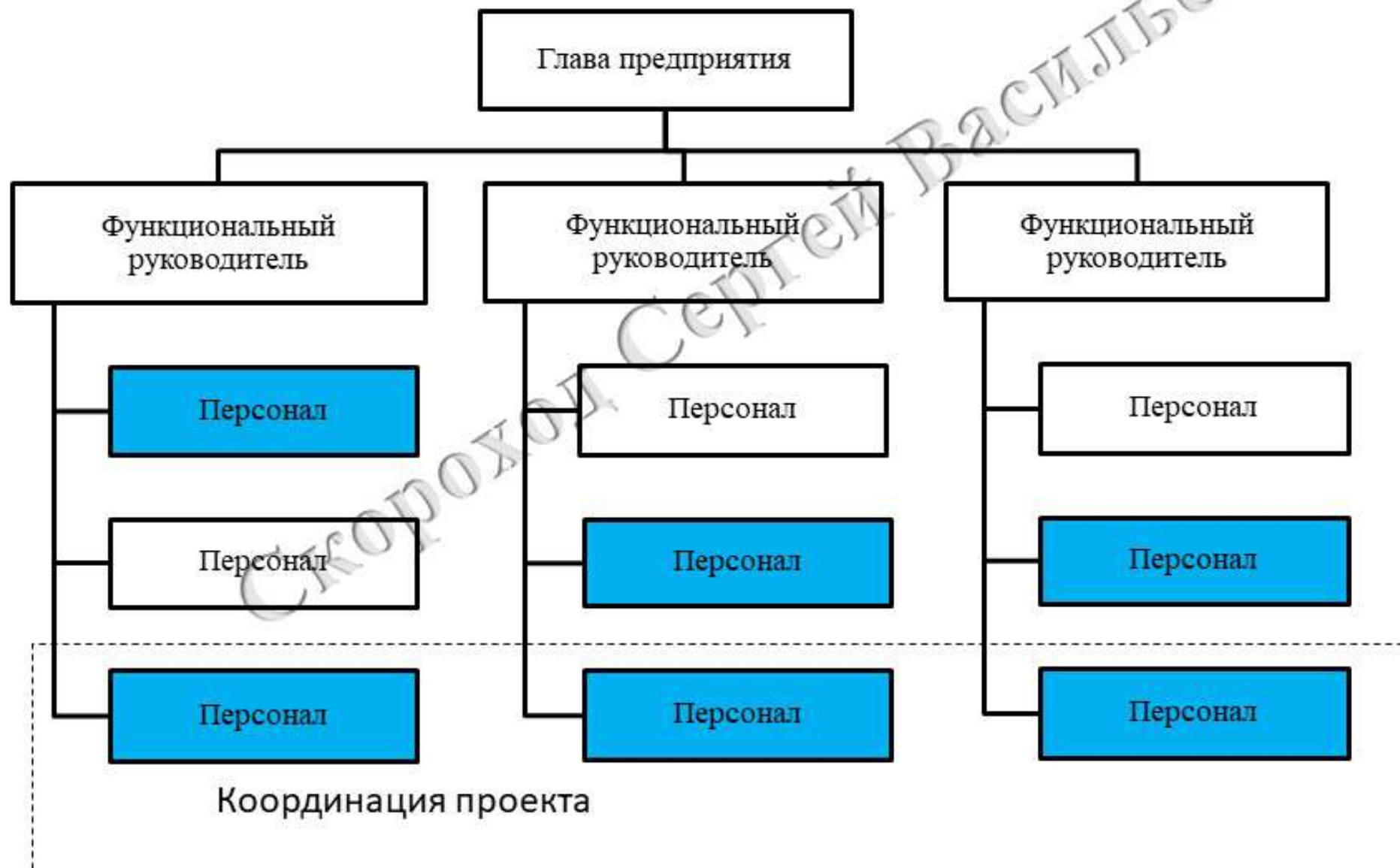


Проектная организация

- В проектной организации члены команд часто собраны в одном месте.
- Большая часть ресурсов организации задействована в работах проектов.
- Менеджеры проектов в значительной степени независимы и обладают большими полномочиями.
- Проектные организации часто имеют подразделения, называемые отделами, но эти подразделения подотчетны непосредственно менеджеру проекта.
- Недостатки: работники работают по временным трудовым договорам и по окончании работ по проекту могут покинуть организацию вместе с накопленным опытом.

Слабая матричная организация

Руководитель проекта разделяет с функциональными руководителями (руководителями отделов) ответственность по заданию приоритетов и управлению работой лиц, назначенных на исполнение проекта



Слабая матричная организация

- Сохраняет многие характеристики функциональной организации.
- Функции менеджера проекта в ней скорее соответствуют функциям координатора или диспетчера проектов, а не менеджера.
- Координация работы сотрудников различных отделов выполняется уполномоченными координаторами этих отделов.
- Окончательные решения принимаются функциональными руководителями по представлению соответствующих координаторов.

Сильная матричная организация

Руководитель проекта разделяет с функциональными руководителями (руководителями отделов) ответственность по заданию приоритетов и управлению работой лиц, назначенных на исполнение проекта

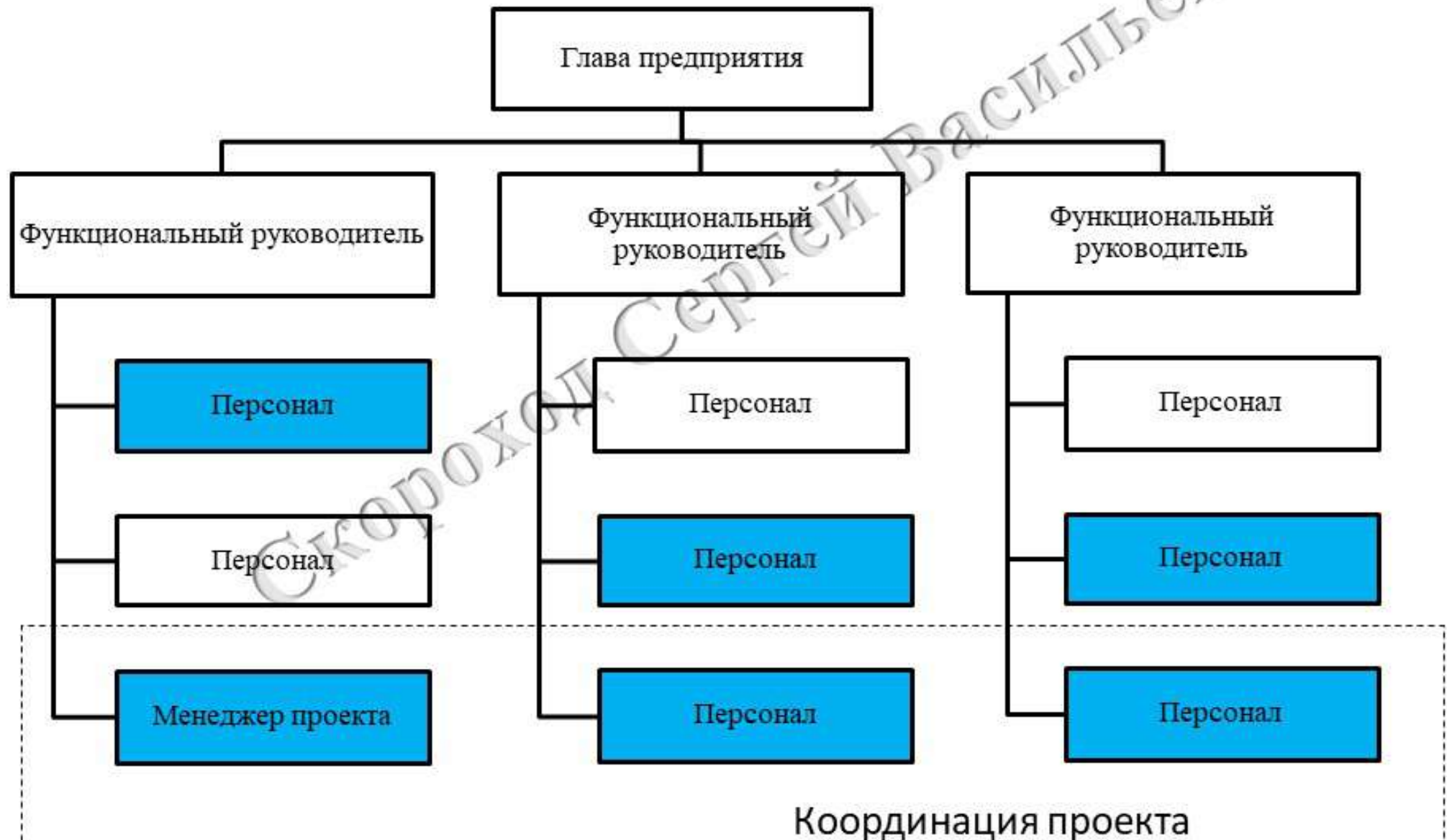


Сильная матричная организация

- Обладает многими характеристиками проектных организаций.
- В ней могут быть штатные менеджеры проектов с широкими полномочиями и входящий в штат управленческий персонал проектов.
- В проекте участвует персонал, подчиненный различным функциональным руководителям.
- Функциональный руководитель отвечает не за вопросы проекта, а за вопросы организации работ.

Сбалансированная матричная организация

Руководитель проекта разделяет с функциональными руководителями (руководителями отделов) ответственность по заданию приоритетов и управлению работой лиц, назначенных на исполнение проекта



Сравнение матричных организаций

Характеристики проекта	Виды организационных структур				
	Функциональная	Матричные			Проектная
		Слабая	Сбалансир	Сильная	
Полномочия менеджера проекта	Незначит. или нет	Ограничено	Средний уровень	Высокий уровень	Полный контроль
Наличие ресурсов	Незначительно или нет	Ограничено	Средний уровень	Высокий уровень	Полный контроль
Контроль бюджета проекта	Функцион. рук.	Функцион. рук.	Смешан.	Менеджер проекта	Менеджер проекта
Роль менеджера проекта	Частичная занятость в проекте	Частичная занятость в проекте	Полная занятость в проекте	Полная занятость в проекте	Полная занятость в проекте
Администр. персонал проекта	Частичная занятость в проекте	Частичная занятость в проекте	Частичная занятость в проекте	Полная занятость в проекте	Полная занятость в проекте

Процессы управления человеческими ресурсами

1. **Планирование человеческих ресурсов** – определение и документальное оформление ролей, ответственности и подотчетности, а также создание плана обеспечения проекта персоналом.
2. **Набор команды проекта** – привлечение человеческих ресурсов, необходимых для выполнения проекта.
3. **Развитие команды проекта** – повышение квалификации членов команды проекта и укрепление взаимодействия между ними с целью повышения эффективности исполнения проекта.
4. **Управление командой проекта** – контроль за эффективностью членов команды проекта, обеспечение обратной связи, решение проблем и координация изменений, направленных на повышение эффективности исполнения проекта.

Форматы документирования ролей



Иерархическая организационная диаграмма

Матрица ответственности				

Матричная диаграмма ответственности

Роль	
Ответственность	
Полномочия	

Текстовый формат

Матрица ответственности

- Показывает ресурсы проекта, выделенные для каждого пакета работ.
- Используется для отображения связей между пакетами работ или операциями и членами команды проекта.

Проектные работы	Проектные роли	Руководитель проекта	Бизнес-менеджер	Менеджер по конфигурации	Администратор	Бизнес-аналитик	Системный архитектор	Разработчик	Тестировщик	Менеджер по качеству
Этап «Анализ»										
Управление программой		Исп. / R								
Планирование проекта		Исп. / R	Согл. / C							
Управление рисками и дефектами		Исп. / R		Исп. / R	Н. / I					Исп. / R
Управление коммуникациями		Исп. / R	Согл. / C		Н. / I					
Управление предложениями		Исп. / R	Согл. / C							
Управление качеством		Исп. / R								Исп. / R
Обучение		Упр. / A			Н. / I	Исп. / R	Исп. / R			
Анализ бизнес-процессов		Упр. / A				Исп. / R				
Требования и конфигурация		Упр. / A		Согл. / C		Исп. / R	Исп. / R			
Сбор бизнес-требований						Исп. / R				
Анализ соответствия решения задачи				Согл. / C		Исп. / R	Исп. / R			
Написание дополнительного кода			Упр. / A					Исп. / R		
Тестирование и обеспечение качества			Упр. / A						Исп. / R	Исп. / R
Инфраструктура			Упр. / A				Исп. / R			
Интеграция и интерфейсы		Упр. / A					Исп. / R	Исп. / R	Исп. / R	
Миграция данных		Упр. / A					Исп. / R	Исп. / R		

Матрица подотчетности

- Определение отношений подотчетности между членами команды.

		Получают отчёт							
Готовят отчёт	Руководитель проекта	Бизнес-менеджер	Менеджер по конфигурации	Администратор	Бизнес-заказчик	Системный архитектор	Разработчик	Тестировщик	Менеджер по качеству
Руководитель проекта	-	по необходимости	по необходимости				никогда	никогда	по необходимости
Бизнес-менеджер	по необходимости	-	никогда				никогда	никогда	никогда
Менеджер по конфигурации	еженедельно	еженедельно	-				по необходимости	по необходимости	по необходимости
...									
Разработчик	по необходимости	по необходимости	по необходимости				-	по необходимости	по необходимости
Тестировщик	по необходимости	по необходимости	по необходимости				по необходимости	-	по необходимости
Менеджер по качеству	еженедельно	еженедельно	по необходимости				по необходимости	по необходимости	-

Потребность работ в ресурсах

- Формируется на основании расписания проекта.

№ этапа	Описание задачи	Оценка трудоемкости, человеко-дни	Роль исполнителя
1. Анализ	1.1.0 Управление программой	26 дней	Руководитель проекта
	1.1.1 Планирование проекта	4 дня	Руководитель проекта, менеджер по качеству
	1.1.2 Управление рисками и дефектами	8 дней	Руководитель проекта, менеджер по конфигурации, менеджер по качеству
	1.1.3 Управление коммуникациями	7 дней	Руководитель проекта, менеджер по конфигурации
	1.1.4 Управление предложениями	4 дня	Руководитель проекта, менеджер по конфигурации
	1.1.5 Управление качеством	3 дня	Руководитель проекта, менеджер по качеству
	1.2.0 Обучения	3 дня	Менеджер по конфигурации, системный архитектор, администратор проекта
	1.3.0 Анализ бизнес-процессов	1 день	Менеджер по конфигурации
	1.4.0 Требования к конфигурации	7 дней	Менеджер по конфигурации, системный архитектор
	1.4.1 Сбор бизнес-требований	2 дня	Менеджер по конфигурации
	1.4.2 Анализ соответствия решения задаче	3 дня	Менеджер по конфигурации
	1.5.0 Написание дополнительного кода	1 день	Разработчик
	1.6.0 Тестирование и обеспечение качества	2 дня	Тестирующий, менеджер по качеству
	1.7.0 Инфраструктура	3 дня	Системный архитектор
	1.8.0 Интеграция и интерфейсы	2 дня	Системный архитектор, тестирующий, разработчик
	1.9.0 Миграция данных	12 дней	Системный архитектор, тестирующий, разработчик
	1.9.1 Семинар по миграции данных	3 дня	Системный архитектор, тестирующий

Реестр навыков

- Формируется для каждого класса персонала.

Реестр навыков для персонала проекта, класс: руководитель проекта		
Категории и компоненты навыков	Рейтинг критичности (от 1 до 4)	Рейтинг способностей (от 1 до 4)
Технические навыки (категория I)		
Способность управлять проектом и его технологией	4	3
Оказание помощи в разрешении проблем	4	4
Взаимодействие с техническим персоналом	3	3
Облегчение достижения компромиссов	4	2
Поощрение новаторского окружения	4	3
Интеграция технических, деловых и человеческих целей	4	1
Способность к системному видению	4	3
Понимание инструментов и методов поддержки инжиниринга	3	4
Административные навыки (категория II)		
Способность к эффективному общению (устному и письменному)	3	4
Оценивание ресурсов и ведение переговоров с целью их получения	4	1
Планирование и организация многофункциональных программ	2	1
Календарное планирование многодисциплинарных операций	4	2
Навыки межличностного общения и лидерства (категория III)		
Способность управлять в неструктурированной рабочей среде	4	3
Оказание помощи в решении проблем	3	4
Построение многофункциональных команд	3	3
Обеспечение вовлеченности персонала на всех уровнях	3	3
Управление конфликтами	4	3
Мотивация людей	4	2
Стратегические навыки (категория IV)		
Способность работать в условиях рисков и неопределенностей	4	2
Стратегическое мышление, планирование и применение решений	4	3

Результаты планирования человеческих ресурсов

1. Распределение ролей и ответственности.

- **Роль.** Обозначение участка проекта, за выполнение которого несет ответственность данное лицо. Для каждой роли должны быть четко определены полномочия, ответственность и границы.
- **Полномочия.** Право задействовать ресурсы проекта, принимать решения и утверждать одобрение действий или результатов.
- **Ответственность.** Работа, которую член команды проекта должен выполнить для завершения операций проекта.
- **Квалификация.** Навыки и способности, требуемые для выполнения операций проекта.

Результаты планирования человеческих ресурсов

2. **Организационная диаграмма проекта.** Это графическое представление состава команды проекта и отношений подотчетности между ее членами.

