# Что такое IT-проект?

**:info:IT-проект, ИТ-проект** — это ограниченное во времени усилие по созданию уникального (в той или иной степени) программного продукта или услуги с использованием информационных технологий.

*:info:***Project Management Body of Knowledge, PMBoK** (англ. «свод знаний по управлению проектами») — руководство, в котором описывается суть управления проектами в терминах интеграции и взаимодействия процессов, а также цели, которым они служат. Считается одним из основных руководств в области управления проектами. PMBoK используется в качестве справочного материала и руководства для своих программ международным Институтом управления проектами (Project Management Institute, PMI).

*:info:***Проект** — это временное предприятие, направленное на создание уникального продукта, услуги или результата (*определение из PMBoK*).

**Эффект от IT-проекта**

Цели, результаты и условия осуществления IT-проектов в каждой компании различаются. Но любой IT-проект через определенное время должен принести компании прибыль, то есть какой-либо положительный эффект. И его не всегда можно напрямую выразить в деньгах.

Например, положительным результатом внедрения IT-проекта может быть экономия времени: в управлении клиентами, управлении знаниями, проведении обучения.

**В чем отличие IT-проекта от обычного проекта?**

1. **Разная структура затрат**  
   В IT-проектах бóльшая часть затрат приходится на рабочую силу.
2. **Интеллектуальность**  
   IT-проект интеллектуально емкий, т.к. разработка связана с интеллектуальной деятельностью разработчиков.
3. **Отсутствие четкого понимания результата**  
   Результат IT-проекта, например, внедрения веб-приложения, неосязаем — его нельзя измерить в килограммах или метрах, его нельзя потрогать. Поэтому требования к результатам проекта, как и план работ, должны быть максимально детальными. Их важно согласовать с заказчиком, стейкхолдерами и командой проекта.
4. **Объем проблем**  
   IT-проекты сталкиваются с уникальными технологическими вызовами, связанными с техническими устройствами, операционными системами или проблемами с базами данных. Также управление усложняется развитием новых технологий и постоянным повышением требований к качеству IT-продукта.

**Виды компаний и их IT-проекты:**

|  |  |
| --- | --- |
| Продуктовая компания | Занимается разработкой собственных IT-продуктов. Продуктом может быть все что угодно: социальная сеть, среда разработки, виртуальный офис. Примеры: VK, Yandex, 1С, 1Т. |
| Аутсорсинговая компания | Занимается разработкой IT-проектов под заказ для других фирм. |
| Консалтинговая компания | Занимается внедрением готовых IT-продуктов, например, системы электронного документооборота. |
| IT-интегратор | Компания, которая проектирует и внедряет в организациях комплексные решения для автоматизации бизнес-процессов, а затем связывает их друг с другом. Интеграция информационных систем — это установка связей между информационными системами организации для получения единого информационного пространства и поддержки сквозных бизнес-процессов в организации. |
| Стартап | Молодая быстро развивающаяся компания, в основе которой лежит инновационная бизнес-идея или технология. |

**Что такое стартап-проект?**

:info: **Startup, стартап** — начинающая компания.

**Стартап** — это коммерческий проект на начальных этапах своей деятельности, который призван решить ту или иную задачу при помощи разных цифровых технологий, сайтов, мобильных приложений.

Создателем IT-проекта, который в будущем будет стоить миллиард, может стать абсолютно любой человек.

Основная задача стартапа — найти важную проблему и предложить самый дешевый и удобный способ ее решения.

Подавляющее большинство стартапов в ИТ сейчас запускают не программисты, а специалисты с узкой квалификацией в других отраслях: врачи, учителя, продавцы, инженеры и так далее. То есть те люди, которые увидели ту или иную проблему и придумали, как ее решить.

# Особенности проектов

Проекты, которые вам предстоит создать на нашем курсе, направлены на командную разработку сайта или веб-приложения. Это популярный, понятный и знакомый формат интернет-ресурса.

Веб-проекты можно разделить по тематике:

* интернет-магазины;
* контентные порталы;
* доски объявлений;
* социальные сети;
* блоги;
* корпоративные сервисы;
* сайты организаций.

**Примеры веб-проектов:**

* Интернет-магазины и маркетплейсы: Ozon. Ozon это маркетплейс, то есть «магазин магазинов». Ozon не только продаёт собственные товары, но и предоставляет платформу для размещения и продажи товаров тысячам других продавцов. Благодаря этому он стал вторым по обороту онлайн-магазином России, в котором содержится 80 млн товарных наименований.
* контентные порталы, блоги: habr.com. Контентным называют сайт, построенный в формате тематических коллективных блогов (или «хабов») с элементами новостного сайта. Habr.com — сайт для публикации новостей, аналитических статей, мыслей, связанных с IT. На таком сайте большую часть контента создают сами пользователи: пишут свои статьи, оставляют комментарии.

:info:**Content, контент** — содержимое, информация, данные.

* доски объявлений: avito.ru. Интернет-сервис для размещения объявлений о товарах, недвижимости, вакансиях и резюме на рынке труда.
* социальные сети: ВКонтакте. Популярная среди русскоязычных пользователей социальная сеть. Она позволяет отправлять друг другу сообщения, создавать собственные страницы и сообщества, обмениваться разным контентом.
* корпоративные сервисы: [Простой Бизнес](http://prostoy.ru/). Cервис для комплексного управления бизнесом: проектами, клиентами, финансами, счетами и продажами.
* сайты организаций: [1t.ru](https://1t.ru/). Наша организация 1Т разрабатывает и внедряет цифровые технологии, ИТ-решения в приоритетных отраслях экономики и социальной сферы. В данный момент мы проводим курсы для получения новой профессии в ИТ.

**Что такое web-приложение?**

**:info:Web, веб** — Всемирная паутина (WWW, World Wide Web).

Web-приложение — программное обеспечение или программа, которую можно открыть с помощью любого браузера (Chrome, Яндекс.Браузер, Safari и т.д.) и практически на любом устройстве (смартфоне, ноутбуке, компьютере). Вебом мы называем ту часть интернета, которая состоит из сайтов. Сайты — то то, что мы воспринимаем через браузеры или через браузероподобные окошки в мобильных приложениях.

:info:**Frontend (фронтенд, клиентская часть)** — часть веб-приложения, которая загружается в браузер пользователя и с которой он взаимодействует. Фронтенд содержит видимые данные (тексты, картинки) и интерфейс (кнопки, поля).

:info:**Backend (бэкенд, серверная часть)** — часть, которая исполняется на веб-сервере. Она готовит в нужном формате данные для клиентской части, получает от неё информацию (например, поля заполненной формы).

Часто мы называем некоторые приложения сайтами, но у web-приложения и web-сайта разные функции и особенности разработки.

**Чем web-приложение отличается от сайта?**

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Web-сайт** | **Web-приложение** |
| Назначение | Позволяет просматривать данные | Позволяет манипулировать данными, взаимодействовать с пользователем |
| Аутентификация, авторизация  *(регистрация, вход, права доступа)* | Не требуется, все данные доступны для просмотра всем | Обычно требуется аутентификация |
| Интерактивность | Пользователь может читать информационный контент, однако не может его менять | Фрагменты страницы генерируются и обновляются «на лету», то есть сразу после взаимодействия с пользователем |
| Сложность разработки | Проще в разработке, меньше настроек, используются системы управления контентом (CMS) | Требуется сложная разработка, дает больше настроек и функций для пользователя |

**Преимущества web-приложений и сайтов в сравнении с мобильными приложениями:**

* Не требуют установки на устройство пользователя.
* Не требуют обновлений, потому что обновляется централизованно.
* Пользоваться можно с любого устройства, на котором есть браузер.
* Не зависят от платформы и операционной системы (ОС): если web-приложение совместимо с браузером — оно работает.
* Разработчику не требуется создавать клиентские приложения для разных ОС, потому что на каждой ОС уже есть браузер.

# Этапы ЖЦ

:info:**Project Life cycle, жизненный цикл проекта, ЖЦ** — это последовательность этапов, через которые проходят проекты от инициации до завершения.

Жизненный цикл определяет рабочий процесс. Понимание жизненного цикла позволяет организовать эффективную работу команды.

:info:**Workflow, воркфлоу, рабочий процесс** — это набор стандартных последовательных действий, которые выполняются на постоянной основе в компании или в команде для достижения определённых целей, например, для реализации проекта.

Есть различные подходы, рекомендации известных авторов и экспертов, но в целом процесс разработки сводится к **нескольким этапам:**

1. Определение концепции.
2. Планирование и подготовка.
3. Проектирование.
4. Реализация и тестирование.
5. Внедрение.
6. Эксплуатация и развитие.

Несмотря на последовательное представление этапов, задачи каждого этапа **часто выполняются одновременно**, в большем или меньшем объёме, в зависимости от степени готовности продукта и других факторов.

Выполнение задачи, проведение мероприятия в хорошо построенном процессе приводит к появлению **артефакта** или части продукта.

:info:**Артефакт** — это объект, созданный в процессе работы: документ, схема, макет, задача, компонент, код и т.д.

Например, к артефактам относятся:

* результаты обсуждения, которые фиксируются в протоколе или заметках;
* проведённые интервью, которые сохраняются в аудио- или текстовом формате;
* процесс планирования, который сопровождается созданием графика, расписания;
* изучение клиентского пути, оформленное в виде визуальной карты;
* результат проектирования архитектуры, выраженный в виде схемы, модели;
* код, который создается в процессе программирования;
* результаты тестирования в виде отчёта с описанием ошибок и т.д.

Разберем по отдельности каждый из этапов жизненного цикла проекта.

1. **Определение концепции**

Предполагается, что сама идея продукта к моменту старта проекта уже существует. Основное внимание на этом этапе уделяется уточнению целей, проверке гипотез, анализу рынка, поиску оптимального решения проблемы. В ходе определения концепции мы выполняем **следующие задачи**:

**1) Определение стейкхолдеров**

С кем говорить о проекте? Кто влияет на проект? На данном этапе надо определить стейкхолдеров, то есть заинтересованных лиц, участников, которые влияют на реализацию проекта, владеют нужной информацией, участвуют в разработке и т.д.

**2) Сбор информации, создание словаря**

Что говорят участники? Что важно знать для реализации? Какую терминологию надо использовать? Чтобы собрать всю необходимую информацию для последующей работы, рекомендуется использовать различные методики: от интервью до обсуждения набросков будущей системы. В ходе сбора информации необходимо понять проблемы и потребности, условия для разработки продукта, текущее состояние дел и возможности, потенциальные риски, процесс, структуру компании, нормативные документы, правила, ограничения, пожелания, ожидания и так далее. Уже на этапе определения концепции формируется словарь, который уточняется по мере детализации. Важно поддерживать его актуальность и полноту.

**3) Определение потребности**

Какая существует проблема? Каковы причины ее появления? Из всей собранной информации необходимо выявить основную потребность или главную проблему, которую нужно решить.

**4) Исследование целевой аудитории**

Кто наши пользователи? Целевая аудитория, пользователи входят в группу заинтересованных лиц и изучению требуется уделить особое внимание, особенно если их лояльность будет влиять на успех продукта.

**5) Изучение пути клиента/пользователя**

Какие действия пользователи совершают для решения проблемы? Какие у них ожидания? Нужно изучить действия пользователя, понять текущие шаги, его путь до знакомства с продуктом, во время использования и после.

**6) Анализ рынка и конкурентов**

Есть ли конкуренты или аналоги? Как решают проблему конкуренты? Проектируя будущую систему, важно изучить отрасль, конкурентов, хорошие решения, аналоги и сделать выводы.

**7) Описание образа продукта**

Что мы будем делать? В результате сбора и анализа данных должно появиться общее описание продукта, его образ, видение, концепция.

**8) Определение целей**

Каковы наши цели? Конечно, цели закладываются на этапе идеи, но по мере анализа информации цели оформляются в более точные формулировки.

**9) Описание функций**

Какие функции нужны пользователю? Какие функции должны быть в системе? Понимая потребности пользователя, ситуацию на рынке и собственные цели, мы можем сформулировать, какие функции должны быть в системе.

**10) Принятие решения и заключение договора**

Делаем ли проект? Этот вопрос необходимо задавать себе и команде на каждом шаге (особенно на старте, когда взвешиваются все за и против). На курсе мы, конечно, рассматриваем вариант, когда решение о разработке положительное — делаем. Однако исследование иногда приходит к выводу о том, что в этом конкретном случае лучше наоборот, не делать. Этот шаг подразумевает также закрепление отношений заказчика и исполнителя в договоре и оформление прочих юридических документов.

1. **Планирование и подготовка**

К моменту планирования и подготовки уже есть понимание, какой продукт мы хотим разработать. Но до старта нужно также оценить возможности по дальнейшей разработке. В ходе этого этапа мы получаем ответы на следующие вопросы:

**1) Управление бэклогом**

Что в итоге нужно сделать? Для оценки и расчётов полезно все функции фиксировать в виде пользовательских требований. Например, можно составить список пользовательских историй. И если на этапе концепции было важно охватить все пожелания пользователей и других заинтересованных лиц, то на этапе планирования нужно расставить приоритеты и обозначить границы.

**2) Оценка трудозатрат**

Сколько нужно времени на разработку? На данном этапе сложно давать точную оценку, однако уже можно обозначать приблизительные трудозатраты для расчёта расходов.

**3) Обозначение границ и версий**

Что сделаем в первую очередь? Реализацию проекта можно разделить на логические отрезки. В гибких методологиях рекомендуется выделять минимальный объём функций для выпуска первой версии. Это необходимо для итерационного выпуска продукта и скорейшего получения обратной связи.

**4) Планирование бизнес-модели**

Сколько стоит разработка? Какие ресурсы доступны? Как продавать продукт? Время рассчитать бюджет. Имея оценку трудозатрат, можно посчитать затраты на реализацию. Также стоит учитывать прочие расходы, в том числе на будущее внедрение, продвижение и поддержку. Если проект подразумевает получение прибыли, то это отличный момент продумать модель монетизации, оценить ёмкость рынка и возможную окупаемость.

**5) Оценка и управление рисками**

Что может пойти не так? Далее важно оценить риски, возможные последствия и необходимые действия для уменьшения риска, продумать шаги в случае наступления рисков.

**6) Выбор модели/методологии**

Как будем разрабатывать? Обычно к этому моменту уже просматривается подход к реализации. Происходит выбор модели, методологии разработки, определяются роли, планируются мероприятия. По необходимости проводится обучение, консультации.

**7) Создание команды**

Кто будет разрабатывать? Конечно, последовательность шагов будет зависеть от проекта и стартовых условий автора идеи. На этом этапе команда может уже активно включаться в работу, либо может стартовать набор необходимых участников.

**8) Организация работы**

Что по условиям работы? Какие нужны инструменты? Важные организационные вопросы: офис, оформление, ПО и прочие инструменты для выполнения задач, проведения коммуникаций и так далее.

**9) Формирование плана работ**

Какой план? Сейчас у нас есть уже не просто идея, а целая концепция продукта (сайта, приложения) с описанием функций, которые мы расставили в порядке приоритетности. У нас уже есть целый офис жаждущих работать специалистов. Нужен план или дорожная карта с описанием полномочий работников, конкретных шагов по достижению целей, паттернов поведения в случае непредвиденных ситуаций.

**10) Постановка задач**

Что конкретно делать каждому? Разные команды по-разному могут выполнять переход от требований к конкретным задачам исполнителя. Важно изначально договориться (или принять правила компании), какого размера будут задачи, какие формулировки будут использоваться, как будут следить за прогрессом и так далее.

1. **Проектирование (спецификация, архитектура)**

На самом деле, проектирование началось уже с того момента, когда мы зафиксировали ожидаемые функции. Описывая требования бизнеса и требования пользователя, изучая проблематику и рынок, мы уже формируем образ будущего продукта. Однако теперь нам надо этот образ детализировать. В рамках этого этапа необходимо определиться со следующими вопросами:

**1) Описание ролевой модели**

Как пользователи отражены в ролях? Изучение пользователей даёт нам понимание, какие роли должны быть в системе и с какими ограничениями.

**2) Описание бизнес-процесса**

Каков бизнес-процесс? Случается так, что для определения функций необходимо сравнить текущий бизнес-процесс as is (как он есть) и будущий бизнес-процесс to be (как будет), например, с учётом автоматизации какого-либо отрезка процесса. Эта задача опциональна, но она может быть полезной.

**3) Моделирование вариантов использования**

Какие действия нужны в системе? Тоже опциональная, но полезная задача — моделирование вариантов использования. Описание вариантов и сценариев использования помогают более детально рассмотреть взаимодействие человека с системой.

**4) Создание наброска**

Как система примерно будет выглядеть? Не обязательно, но полезно сделать набросок. Уровень детализации наброска зависит от целей. Его можно сделать еще на этапе концепции и продемонстрировать заинтересованным лицам для обсуждения.

**5) Детализация требований**

Какие конкретно требования? Для оценки и планирования достаточно было обозначить функции, дополнить схемой или наброском. Но для того чтобы спроектировать конкретное техническое решение, продумать все детали, выбрать нужные технологии и корректно выполнить задачи, нужно детально описать задание. Описание может оформляться в спецификацию или критерии приёмки пользовательских историй.

Требования — не просто «хотелки» всех желающих высказаться и повлиять на проект. Ответственные за описание требований должны постоянно проверять полноту, взаимосвязи, корректность, влияние на цели и т.д. Помимо функциональных возможностей, нужно описывать требования к безопасности, надежности, скорости работы и т.д.

**6) Проектирование UI/UX**

Как будет выглядеть система? Какое будет ее поведение? Над пользовательским интерфейсом мы задумываемся с момента появления идеи, отражаем его в наброске. Далее, уже с пониманием своей целевой аудитории и всех шагов пользователя, продумываем его путь в системе и внешний вид самой системы. Иногда речь может идти о дизайне новых устройств, звуков, принципа работы.

**7) Моделирование требований**

Как посмотреть на систему до реализации? Требования — это не только перечисления функциональных возможностей, а пользовательский интерфейс — это не вся система. Чтобы отразить максимальное количество уровней системы, мы моделируем требования, используя разные виды диаграмм. Например, рассматриваем отдельно модель данных, статусы документов и пр. Бизнес-процесс, о котором шла речь раньше, — это тоже модель.

**8) Проектирование архитектуры**

Как будет устроена система? Из каких частей? Какие технологии использовать? Требования, макеты, диаграммы позволяют продумать структуру, внутреннее устройства системы, возможные интеграции, а также выбрать технологии, инструменты, языки программирования и пр.

1. **Реализация и тестирование**

Часто руководства по проджект-менеджменту объединяют этапы проектирования, реализации, тестирования и внедрения в один общий этап реализации проекта (между планированием и эксплуатацией). Однако нам важно углубиться в детали, поэтому мы подробно рассмотрели этап проектирования и только сейчас переходим к реализации. В ходе этого этапа мы приближаемся к воплощению.

**1) Конструирование и программирование**

Ранее мы поставили задачи, описали требования и решения, определили внешний вид системы и внутреннее устройство. Теперь — приступаем к созданию продукта. Разнообразие инструментов и технологий позволяют погружаться на разный уровень глубины программирования. Но любой процесс программирования сопровождается контролем качества реализации. Помимо тестирования самого продукта, проверяется код, соответствие соглашениям, лучшим практикам и т.д.

**2) Планирование тестирования**

Как будем тестировать? На основе требований разрабатывается план тестирования, который может включать не только прохождение по сценариям пользователя, но и, например, проверять устойчивость при большой нагрузке, скорость ответа с сервера, количество ошибок пользователя при выполнении операции и пр.

**3) Выполнение тестирования**

В результате тестирования фиксируются отчёты, замечания, проблемы, ошибки. Важная задача по улучшению качества тестирования (и как следствие, качества продукта) — автоматизация тестирования.

1. **Внедрение**

Внедрение тесно связано с разработкой и эксплуатацией. Если это заказная разработка, то на этом этапе можно наблюдать, как команда разработки передает продукт заказчику, сопровождая его необходимой документацией. В ходе этого этапа решаются следующие задачи:

**1) Подготовка инфраструктуры, выполнение поставки, ввод в эксплуатацию**

Куда и как разворачивать систему? Обычно процесс интеграции, поставки и развёртывания начинается одновременно с программированием. Однако это необходимо только при наличии сложной архитектуры, частых релизов и т.п. В ином случае данная задача может быть решена разработчиком с помощью специализированных инструментов (например, если в вашем проекте по созданию сайта-визитки задействован один разработчик). Необязательно ждать завершения проекта для отслеживания прогресса. Для выполнения тестирования и проведения демонстрации могут создаваться дополнительные окружения и стенды (помимо промышленного).

**2) Демонстрация системы**

Демонстрация проводится время от времени по мере разработки. Это помогает проверить, совпадают ли ожидания с реальностью. Современные методологии рекомендуют организовывать процесс именно так, чтобы автор идеи и все заинтересованные лица могли наблюдать прогресс.

**3) Подготовка руководства пользователя**

До полноценного выпуска система должна проверяться на соответствие законам. С пользователем (особенно в случае наличия платных функций) заключается особое соглашение (то самое, которое никто не читает перед установкой).

**4) Завершение разработки, оформление закрывающей документации**

Попадание продукта в промышленную эксплуатацию в принципе означает окончание разработки. Это значит, что пора завершить работу и с административной точки зрения, в том числе подписать закрывающие документы.

1. **Эксплуатация и развитие**

Кажется, что пора радоваться и открывать шампанское. Однако далеко не для всех проектов ввод в эксплуатацию является конечной точкой. Стоит вернуться к бизнес-модели, которую мы описывали на старте, освежить план продвижения и приступить к его выполнению. А также обеспечить мониторинг, организовать поддержку и пообщаться с теми пользователями, для кого предназначается продукт. В ходе этого этапа мы уходим от выполнения задач в цикл развития и задаем новые вопросы:

**1) Продвижение и продажи**

На этапе продвижения выполняется огромный спектр работ, связанных с рекламой, маркетингом и монетизацией.

**2) Технический мониторинг**

Мониторинг работы системы желательно начинать со старта разработки. Важные нагруженные системы, для которых заявляются определённые характеристики и работоспособность которых влияет на прибыль, нуждаются в дополнительных инструментах контроля состояния. Для побочных систем эта задача необязательна.

**3) Поддержка и развитие**

Что говорят пользователи? На что жалуются? Что нравится? За что платят? Сбор информации для развития продукта происходит из множества источников. Можно продолжить коммуникацию с заинтересованными лицами, можно исследовать рынок, можно общаться напрямую с пользователями, можно собирать статистику использования, можно пересмотреть цели и изменить продукт.

А можно остановиться и порадоваться успеху.

# Дополнительно

**Типы и классификация ИТ-проектов**

**Выделяют три типа ИТ-проектов:**

1. **Проекты разработки ПО (software development projects)**
2. **Проекты внедрения ПО (software implementation projects)**
3. **ИТ инфраструктурные проекты (IT Infrastructure projects)**

**Первый тип – это проекты разработки программного обеспечения (software development projects). Программный продукт или программа – это всегда код, который работает с информацией. Под «работает» понимается, что программа информацию создает, обрабатывает, хранит или делает все сразу. Это информационный продукт.**

**Цель таких ИТ проектов – создать конкретное веб-приложение по требованию заказчика, которое принесет ценность для его бизнеса. Например, создать веб-приложение для учета финансов компании, или разработать электронный магазин одежды, или текстовый редактор.**

**Второй тип – это проекты внедрения программного обеспечения (software implementation projects). Это когда уже разработанное приложение внедряется в готовую инфраструктуру заказчика. Такие проекты обычно включают целый набор интеграций с другими информационными системами заказчика, адаптацию под его реальные бизнес-процессы, последующее изменение регламентов работы и обучение персонала. Примеры таких проектов: внедрение CRM в банке, внедрение финансового модуля SAP в логистической компании, внедрение корпоративной системы онлайн обучения и т. д. Эти проекты всегда тесно связаны с заказчиком, для которого делается внедрение.**

**Третий тип – это ИТ инфраструктурные проекты (IT infrastructure projects), которые характеризуются построением систем взаимодействия программного обеспечения, сервисов и аппаратной части (hardware). Обычно это комплексные проекты, цель которых – создать основу, на которой будут функционировать все ИТ-сервисы организации.**

**Примеры таких проектов: построение ИТ-сервисов для отеля, построение ИТ-инфраструктуры банка и т. д.**

**Признаки классификации ИТ-проектов:**

**– по масштабу;**

**– по длительности;**

**– по отраслевой принадлежности;**

**– по функциональному направлению;**

**– по степени сложности;**

**– по характеру привлечения сторон.**

**Выделяют оперативные и стратегические. Оперативные проекты связанны с нынешней работой объекта инвестирования. Данные проекты представлены малыми и простыми, их финансирование никак не превосходит 100 000 рублей, продолжительностью до месяца и затрагивающие работу не более 10% сотрудников организации.**

**Согласно продолжительности: краткосрочные, среднесрочные, долгосрочные. Краткосрочные проекты включают в себя своевременные проекты предприятия, продолжительностью до 1 года. Среднесрочные проекты обладают продолжительностью до 5 лет и представлены стратегическими проектами предприятия, региональными проектами. Долгосрочные проекты имеют продолжительность свыше 5 лет и представлены мегапроектами различной направленности.**

**Согласно функциональному направлению, акцентируют: финансовые технологические, производственные, исследования и развития, маркетинговые, по управлению персоналом, комбинированные. Финансовые проекты связанны с финансированием объектов, осуществлением всевозможных финансовых операций (кредитных, валютных и т. д.), технологические – с формированием и усовершенствованием технологий (например, внедрение новой производственной линии, разработка программного обеспечения и т. д), производственные – это проекты производственной направленности, проекты исследования и развития – это проекты, связанные с проведением рыночных исследований с целью разработки направлений дальнейшего развития; научно-исследовательская деятельность (например, разработка методик автоматизации бизнес-процессов университета (система электронного документооборота, система дистанционного обучения, информационная система независимого экзаменационного центра)).**

**Согласно уровню сложности: монопроекты, мультипроекты, мегапроекты.**

**Монопроекты – проекты, которые выполняются в рамках одной компании или структуры; преследующие одну и ту же инновационную задачу, исполняются в строгих временных и финансовых рамках (например, нвестиционные, инновационные и др.);**

**Мультипроекты – предполагают собой портфель десятков монопроектов, направленных на формирование научно-технического комплекса, решение крупномасштабной технологической задачи (например, разработка и внедрение внутрифирменных систем и тд);**

**Мегапроекты – многоцелевые комплексные программы, включающие мультипроекты и монопроекты, сопряженные между собой одной системой целей и обладающие централизованным управлением и финансированием. В рамках мегапроектов выполняются подобные цели, как решение федеральных и региональных проблем экологии, увеличение конкурентоспособности российских фирм, их продукций и технологий (например, разработка высокотехнологичного программно-инструментального комплекса для реализации систем управления технологическими процессами).**