



ТЕХНИЧЕСКИЙ РЕГЛАМЕНТ РАЗРАБОТКИ ИНТЕРНЕТ-ПРОЕКТА



ВВЕДЕНИЕ.	2
1. ТЕРМИНЫ И ОБОЗНАЧЕНИЯ.....	3
2. КЛАССИФИКАЦИЯ ИНТЕРНЕТ-ПРОЕКТОВ.	5
3. ЭТАПЫ В РАЗРАБОТКЕ ИНТЕРНЕТ-ПРОЕКТА.	6
3.1. Запрос. Определение целей Проекта и его дальнейшее позиционирование.	6
3.2. Брифинг.....	7
3.3. Подготовка предложения.	8
3.4. Планирование.....	9
3.5. Реализация.....	9
3.5.1. Аналитика	10
3.5.2. Стратегия.....	10
3.5.3. Разработка структуры и создание прототипов страниц.	10
3.5.4. Создание дизайн-концепта главной и внутренней страницы, создание дизайн-макета страниц и их согласование.	11
3.5.5. Составление технического задания (ТЗ) и уточнение ценовой позиции Проекта.	12
3.5.6. Front-end	13
3.5.7. Back-end	13
3.5.8. Тестирование, наполнение контентом.	14
3.5.9. Подготовка к запуску проекта.	15
3.5.10. Перенос проекта на основной домен.....	15
3.5.11. Промышленное тестирование Проекта.	16
3.6. Сдача Проекта. Обучение персонала.	16
3.7. ГАРАНТИИ.....	16
3.8. Изменения проекта после запуска.....	17
4. ТРЕБОВАНИЯ К ИНФОРМАЦИИ, ПРЕДОСТАВЛЯЕМОЙ ЗАКАЗЧИКОМ.	18
5. ПЕРСОНАЛ, ПРИНИМАЮЩИЙ УЧАСТИЕ В ПРОЦЕССЕ РАЗРАБОТКИ ИНТЕРНЕТ-ПРОЕКТА, И ЕГО ОСНОВНОЙ ФУНКЦИОНАЛ.....	19
6. ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К РАЗРАБАТЫВАЕМОМУ ПРОЕКТУ.....	20
7. ТЕХНОЛОГИИ И ИНСТРУМЕНТЫ, ПРИМЕНЯЕМЫЕ ПРИ СОЗДАНИИ ИНТЕРНЕТ-ПРОЕКТОВ.....	23

ВВЕДЕНИЕ.

Настоящий документ составлен на основе анализа процесса разработки и внедрения интернет-проектов ведущими национальными и международными digital-агентствами и призван его стандартизировать.

Технический регламент носит рекомендационный характер и относится к категории индустриальных стандартов.

Основными целями данного документа являются:

- Регламентировать технические и процессные подходы к разработке и внедрению интернет-проектов.
- Определить необходимые технические требования и стандарты к используемой во время разработки интернет-проектов информации.
- Определить требования и квалификацию к участникам процесса разработки интернет-проектов.
- Регламентировать общие и специализированные технические требования к интернет-проектам.

Регламент принят Всеукраинской Рекламной Коалицией в 2018 году

1. ТЕРМИНЫ И ОБОЗНАЧЕНИЯ.

Интернет-проект – определенная последовательность задач, направленная на достижение конкретных целей в течение заданного периода времени, с установленными требованиями к качеству результатов и возможными рамками расхода ресурсов, реализуемая при помощи современных информационных технологий, представленных в Интернет.

Команда проекта – специалисты, задействованные на различных этапах разработки проекта. В зависимости от специфики проекта команда может меняться по составу и уровню квалификации. Обязательным неизменным членом команды проекта есть Project Manager, который осуществляет управление и коммуникацию проекта. В команде проекта обычно находятся как представители Разработчика, так и представители Заказчика.

Заказчик – компания или частное лицо, в интересах которого ведется разработка. На Заказчика в процессе разработки накладываются определенные обязательства и требования к участию в проекте.

Разработчик – компания или частное лицо, представляющее в процессе взаимодействия сторону Команды проекта.

Пользователь – человек, физическое лицо, которое является потенциальным или реальным посетителем разработанного интернет-проекта. Именно его портрет и поведенческие характеристики учитываются при разработке интернет-проекта для обеспечения потребностей, по которым он будет взаимодействовать с проектом.

Редизайн – изменение дизайна интернет проекта без изменения функционала и структуры интернет-проекта

Реинжиниринг – изменение дизайна, функционала и структуры интернет-проекта с сохранением общей концепции и стилистики оформления

Cost – стоимость разработки / реализации отдельной задачи, модуля, этапа или целого интернет-проекта

Rate – стоимость часа работы специалиста со стороны Разработчика. Исходя из данной стоимости высчитывается стоимость реализации отдельной задачи, модуля, этапа или целого интернет-проекта как rate умноженный на количество часов, необходимых для реализации данного проекта.

Техническое задание (ТЗ) — исходный документ на разработку интернет проекта. ТЗ устанавливает основное назначение разрабатываемого проекта, его технические характеристики, показатели качества и технико-экономические требования, этапы выполнения работ и документацию к проекту, а также специальные требования если таковые имеются. Техническое задание является юридическим документом — как приложение включается в договор между Заказчиком и Разработчиком на проведение проектных работ и является его основой: определяет порядок и условия работ, в том числе цель, задачи, принципы, ожидаемые результаты и сроки выполнения. То есть

должны быть объективные критерии, по которым можно определить, сделан ли тот или иной пункт работ или нет. Все изменения, дополнения и уточнения формулировок ТЗ обязательно согласуются с Заказчиком и им утверждаются. Техническое задание, как термин в области информационных технологий – это юридически значимый документ, содержащий исчерпывающую информацию, необходимую для постановки задач на разработку и запуск интернет-проекта.

2. Классификация интернет-проектов.

По типам Проекты делятся на:

- Проекты **e-commerce** (интернет-магазины, витрины, каталоги)
- **Коммуникационные** проекты, направленные на построение общения между Пользователями (форумы, порталы, социальные сети, блоги)
- **Информационные** проекты, направленные на информирование Пользователей (новостные сайты, кулинария, здоровье, спорт, красота и т.д.)
- **Сервисные** проекты (поисковые, аналитика, онлайн-игры и т.д.)

В зависимости от применяемой технологии создания интернет-проектов их можно классифицировать на:

- **Статические** интернет-проекты, содержащие статические HTML или XHTML страницы. Это определенный набор статических файлов (текст, таблицы, рисунки), которые создаются при помощи языка разметки HTML и хранятся в готовом виде в файловой системе сервера.
- **Динамические** интернет-проекты, в которых страницы генерируются в процессе исполнения запроса Пользователя.
- **Интерактивные** интернет-проекты с использованием технологий анимирования (интерактивная векторная анимация для Web).
- **Комбинированные** интернет-проекты, которые сочетают в себе вышеуказанные технологии при разработке интернет-проектов.

По структуре проекта:

- Одностраничные
- Многостраничные

По виду формирования контента:

- Собственный (генерируется администратором интернет-проекта)
- Генерируемый пользователями

По степени и глубине интеграции:

- Без интеграции
- С интеграцией (CRM, ERP, каталоги, биллинговые системы, логистические системы и т.д.)

Система управления Проектом и дизайн:

- Шаблонные и готовые решения, имеющие ряд функциональных ограничений
- Индивидуальные, которые разрабатываются под запрос Заказчика

3. Этапы в разработке интернет-проекта.

Процесс разработки Проекта состоит из следующих основных этапов:



- Определение целей Проекта и его дальнейшего позиционирования.
- Подготовка предложения, переговоры и принятие решения.
- Планирование и техническая реализация Проекта.
- Сдача Проекта и промышленное тестирование.
- Гарантии и техническая поддержка.
- Доработки и усовершенствования Проекта.

3.1. Запрос. Определение целей Проекта и его дальнейшее позиционирование.

Залог успешной и эффективной реализации Проекта заключается в четком понимании и определении его целей. В большинстве случаев, цели Проекта устанавливаются Заказчиком, а потом уточняются или корректируются вместе с Разработчиком с пониманием имеющейся у Заказчика стратегии (бизнес; маркетинг; продукт).

Если Заказчик на данном этапе *четко не понимает для чего ему нужен Проект и какие задачи он должен решить*, то можно с большой уверенностью говорить о том, что в будущем он будет недоволен Проектом (независимо от качества и сроков реализации Проекта), а потраченные ресурсы (время, деньги, энергию) будет считать потерянными.

Основной задачей Разработчика на этапе определения целей Проекта является поверхностный анализ отрасли и ниши, в которой представлен Заказчик, а также уточнение основных задач, которые должен решить интернет-проект. На этом этапе также важно выяснить у Заказчика следующую дополнительную информацию:

1. Профиль и описание целевой аудитории Проекта. Помните – Проект должен решать бизнес-задачи Заказчика путем максимального попадания в целевого Пользователя. Именно от него (от целевого Пользователя) зависит общая структура и наполнение сайта, информационная направленность и digital-tone Проекта. Именно соединение целей Проекта и максимально полное описание целевого Пользователя является основой успешной реализации и внедрения.
2. Конкурентные референсы (ссылки) на другие Проекты (конкуренты по отрасли/нише), которые наиболее близки Заказчику и почему.
3. Референсы на лучшие/передовые Проекты (зачастую не связаны с нишей), которые занимают лидирующие позиции в своей отрасли.

4. Что не устраивает Заказчика в его текущем Проекте (если запрос Заказчика связан с редизайном/реинжинирингом имеющегося Проекта).

После получения всей необходимой информации от Заказчика необходимо выполнить поверхностный анализ сегмента и ниши для того, чтобы:

- Выделить общие тенденции развития/состояния сегмента в интернет
- Определить готовность Заказчика к реализации Проекта
- Определить конкурентные преимущества Проекта

После проведения предварительной аналитики, можно переходить к этапу брифинга.

3.2. Брифинг.

Брифинг - второй и не менее важный этап в определении целей Проекта. Он направлен на:

1. Проверку запроса Заказчика на разработку Проекта и его соответствие истинным целям.
2. Полное и однозначное понимание задачи Заказчика с частичным погружением в его среду (бизнес).
3. Уточнение важных деталей для дальнейшей качественной реализации интернет-проекта.
4. Определение зон ответственности и полномочий каждой из сторон, а также степень и глубину взаимодействия участников команды проекта.
5. Оценку ресурсного состояния Заказчика (наличие у него необходимых для реализации ресурсов).
6. Выяснение критериев успешной реализации и функционирования Проекта – KPI проекта. К таким критериям можно отнести технические критерии Проекта (быстродействие, надежность и т.д.) и поведенческие характеристики пользователей (показатель отказов, среднее время пребывания на сайте, глубина просмотра и т.д.)

Именно от качественно проведенного брифинга зачастую зависит успех реализации Проекта. Не исключено, что после проведения брифинга, заявленная Заказчиком цель в его первичном обращении, может немного или существенно измениться. Иногда брифинг может закончиться тем, что Заказчик осознает свою неготовность к реализации Проекта по внутренним причинам. В таком случае лучше это выяснить на раннем этапе и сохранить ресурсы для будущей разработки.

Другим немаловажным аспектом качественно проведенного брифинга, является четкое понимание и основа для составления коммерческого предложения и временных ориентиров на разработку Проекта.

Желательно на этапе брифинга получить от Заказчика его digital или креативную стратегию (в зависимости от типа Проекта) и бюджет. Если у Заказчика нет четкого бюджета, он может предоставить градацию бюджета, в который он планирует вложиться (10к, 25к, 50к, 100к \$)

3.3. Подготовка предложения.

После того, как Разработчику понятны цели и глубина Проекта, он готовит коммерческое предложение на разработку. Как правило Коммерческое предложение состоит из нескольких основных частей:

- Описание целей Проекта, подтверждение ожидаемых KPI, которые будут достигнуты после разработки
- Креативная концепция: основная идея, референсы или скетчи, которые передают суть концепции
- Структура проекта: основной функционал, разделы, цели для каждой ЦА и основные правила поведения пользователей
- Бюджет и тайминг проекта
- Квалификация Разработчика, которая подтверждает его возможность реализовать данный Проект

Бюджет и тайминг – это документ, который содержит в себе все этапы разработки Проекта, перечень специалистов, задействованных в проекте, количество часов каждого специалиста, тип решаемых ими задач или реализуемый функционал и общий тайминг реализации по этапам.

При составление такого документа возможны два подхода:

- **Этапный** подход - оценка стоимости определенного этапа реализации Проекта (например, дизайн главной страницы) без уточнения участия тех или иных специалистов и затраченного на это времени. При этом указывается общая стоимость реализации этапа и время его реализации.
- **По-часовой** подход – оценка стоимости определенного этапа на основании суммирования стоимости участия каждого отдельного специалиста. В данном случае стоимость участия каждого отдельного специалиста рассчитывается как произведение его часов на его rate.

Детально составленный Бюджет позволяет проверить все ли необходимые этапы Проекта учтены и вовремя их скорректировать при необходимости. Детально составленный Тайминг в привязке к каждому этапу проекта позволяет точно спрогнозировать релиз проекта и исключить неожиданные сдвиги сроков.

При просчетах сложных многофункциональных или нестандартных решений допускается ориентировочный просчет. Детальный просчет и планирование Разработчиком осуществляется на платной основе так как требует глубокого погружения и значительных временных затрат.

3.4. Планирование

Основной целью этапа планирования является обсуждение проекта со всеми стейкхолдерами со стороны Заказчика для уточнения (и возможной корректировки при необходимости) всех нюансов реализации проекта.

Результатом этапа является уточненный бюджет и тайминг проекта, согласованный процесс приема – передачи проекта, обучения представителей Заказчика. Все это прописывается в Договоре либо является приложением к заключенному ранее Договору и подписывается на данном этапе.

В тайминге обычно указываются не только сроки и обязательства Разработчика по предоставлению информации на утверждение Заказчику, а и сроки, и обязательства Заказчика по предоставлению необходимой информации и ответов на вопросы/согласование этапов.

3.4.1. Условия оплаты

Если между Разработчиком и Заказчиком не установлены ранее другие договоренности по оплате, рекомендованы следующие варианты:

- Если срок разработки проекта не превышает 120 рабочих дней, то оплата производится по следующей схеме: 50% от суммы Договора выплачивается в течении 5 дней после подписания Договора, 25% от суммы Договора выплачивается после утверждения дизайн концепции, всех страниц и анимации и Технического задания, 25% выплачивается после сдачи проекта. В случае отсрочки платежа – через 30 дней после сдачи проекта.
- Если срок разработки превышает 120 рабочих дней, рекомендовано составлять график ежемесячных оплат в течении реализации проекта, при условии что Разработчик получает 50% от суммы Договора до начала работ и не менее 75% от суммы договора до сдачи проекта.
- В случае нарушения сроков оплаты со стороны Заказчика, Разработчик может приостановить работы, взяв в реализацию другой проект, чтобы обеспечить команду разработчиков плановой загрузкой. При этом сроки реализации проекта будут установлены заново в момент получения следующего платежа и возобновления работ.

3.5. Реализация

При реализации Проекта можно выделить следующие основные этапы:

1. Аналитика
2. Стратегия
3. Разработка структуры и прототипов страниц
4. Создание дизайн-концепта главной и внутренней страницы, создание дизайн-макета страниц и их согласование
5. Составление Технического Задания (ТЗ) и уточнение ценовой позиции Проекта
6. Front-end
7. Back-end
8. Тестирование, наполнение контентом.
9. Подготовка к запуску проекта.

10. Перенос Проекта на основной домен.
11. Промышленное тестирование Проекта.
12. Сдача Проекта Заказчику, обучение персонала Заказчика.

Если посмотреть глобально на процесс и этапы разработки Проекта, то можно выделить

- Этапы, которые могут идти параллельно и независимо друг от друга. К таким этапам можно отнести:
 - Разработка дизайна и составление ТЗ
 - Тестирование и наполнение контентом
- Этапы, которые идут последовательно и зависят от окончания и полученных результатов на предыдущем этапе. Например, нельзя переходить к этапу Front-end, если у вас не закончен этап дизайна.

3.5.1. Аналитика

Предварительный этап, на котором Разработчик глубоко погружается в тему и продукт Заказчика, изучает рынок, общемировые и локальные тренды, позиционирование конкурентов.

Проводит аналитику продукта/услуги Заказчика. И аналитику по взаимодействию потребителя с аналогичным продуктом на других проектах.

3.5.2. Стратегия

ОСНОВНЫЕ ЗАДАЧИ ЭТАПА:

- Сегментирование ЦА и составление портретов потребителей. Какие задачи решает потребитель при взаимодействии с проектом, какие проблемы и опасения у него есть, какие выгоды ожидает получить.
- Поведенческие характеристики потребителя при взаимодействии с проектом. Помимо этих маркетинговых стратегических данных необходимо выделить поведенческие характеристики потребителя, которые позволят ему наиболее эффективно взаимодействовать с проектом и сделают проект понятным и доступным, соответствующим его ожиданиям. Какие действия на проекте для него привычны, что его раздражает, что ему может не хватать.
- Определение эмоций и мотивации, которую должен испытать потребитель при взаимодействии с проектом и которую, соответственно, разработчик должен донести при помощи визуальной концепции.
- Выбор и обоснование визуальной концепции проекта.

3.5.3. Разработка структуры и создание прототипов страниц.

Задача этого этапа состоит в том, чтобы максимально детально проработать будущую структуру и функционал сайта с учетом его отображения на разных девайсах (смартфоны, планшеты, десктопы). Это позволит на этапе дизайна существенно сократить срок реализации Проекта.

Прототип страницы необходим для:

- Визуализации структуры Проекта с учетом юзабилити и UI/UX дизайна
- Формирование customer journey с учетом и пониманием поведения Пользователя
- Детальная проработка навигационных инструментов Проекта на разных девайсах (смартфоны, планшеты, десктопы)
- Взаимосвязи между разделами и блоками за счет визуализации в интерактивном прототипе.

Прототипы бывают статические – набор схем/структур страниц и интерактивные – на которых пользователь уже может оценить взаимодействие с Проектом.

Рекомендуется предоставлять прототипы Заказчику в виде интерактивных взаимосвязанных страниц для более точного понимания общей структуры.

Перед подачей на утверждение Заказчику рекомендуется провести тестирование по сегментам ЦА по нескольким типам задач, которые необходимо решить.

Перед тестированием необходимо составить до 5 разных задач, которые должны решить для себя пользователи при взаимодействии с проектом, выбрать представителей ЦА и провести тестирование с фиксацией времени решения задачи.

Важно во время проведения тестирования ведущему правильно фиксировать сложные моменты, барьеры и паузы, возникающие при взаимодействии пользователя с системой, подать полученный список для рассмотрения команде проекта и возможной корректировки прототипа перед утверждением Заказчиком.

3.5.4. Создание дизайн-концепта главной и внутренней страницы, создание дизайн-макета страниц и их согласование.

Задача этого этапа состоит в том, чтобы на основании согласованных прототипов Проекта, разработать общую стилистику и концепцию дизайна. А также разработать анимацию если такая предусматривается проектом и стилистику навигационных элементов. После утверждения общей концепции, логики проекта и прототипов всех страниц – разработать дизайн всех страниц. Так же важно на этом этапе продумать:

- эффекты, которые будут применяться при верстке Проекта;
- таблица применяемых стилей для стандартных элементов проекта (H1, H2, таблицы и т.д.);
- продумать индивидуальность Проекта и его отличительные черты: авторские навигационные элементы, авторская анимация, motion дизайн.

По согласованию на этапе планирования дизайн к утверждению может быть представлен в виде общей концепции дизайна и ключевых моментов, в виде дизайна всех страниц для одного типа устройств, выбранного как основное (обычно это десктоп) или в виде дизайна всех страниц для всех типов устройств. Все макеты дизайнов утверждаются уполномоченным представителем Заказчика.

Количество итераций обсуждений, комментариев и корректировок к каждому макету ограничивается двумя раундами взаимодействия с Разработчиком.

3.5.5. Составление технического задания (ТЗ) и уточнение ценовой позиции Проекта.

Документ техническое задание обязательно составляется в случае, если Проект реализуется с использованием каскадной модели («waterfall») ведения проектов. В таком случае в этом документе прописываются все детали функционала, максимально подробно, включая прототипы или дизайны страниц, указаны все элементы и взаимодействие между ними.

Для Проектов, которые реализуются с использованием гибкой модели («agile»), этот документ может быть применен в другой формулировке – при таком подходе ТЗ может составляться на каждую отдельную задачу или набор задач. Или ограничиться просто описанием задач.

ТЗ – основной и самый главный документ во всем этапе разработки и реализации Проекта и его задача свести к минимуму разницу между представлениями и ожиданиями двух сторон от реализации Проекта.

ТЗ регламентирует:

- логику взаимодействия Пользователя с Проектом и Администратора с административной панелью Проекта,
- интеграцию с внешними ресурсами или системами,
- технические требования к Проекту (браузеры, разрешения, скорость загрузки страниц, надежность работы и устойчивость к нагрузкам),
- технические требования к данным.

ТЗ составляет Разработчик в тесном взаимодействии с Заказчиком.

В ТЗ должны быть прописаны четкие и измеримые атрибуты качества работы Проекта (например, скорость загрузки, и т.д.) во избежание в будущем неопределенности и двойного трактования полученных результатов.

Примерная структура ТЗ:

1. Общие положения.
 - a. Глоссарий
 - b. Предмет разработки
 - c. Назначение документа
2. Требования к графическому наполнению Проекта
 - a. Требования к наполнению Проекта
 - b. Порядок согласования и утверждения наполнения Проекта
3. Функциональные требования
 - a. Задачи, которые должен решать Проект
 - b. Профиль целевой аудитории, классы Пользователей
 - c. Структура Проекта, количество и перечень страниц
 - d. Требования к системе управления Проектом
 - e. Требования к разделению доступа
4. Требования к видам обеспечения
 - a. Требования к информационному обеспечению (языковые версии, структура и объем контента)
 - b. Требования к программному обеспечению
 - c. Требования к техническому обеспечению

- d. Требования к эргономике и технической эстетике
 - e. Требования к надежности
 - f. Требования к хостингу
5. Требования к приемке-сдаче Проекта
- a. Требования к наполнению информацией
 - b. Требования к персоналу
 - c. Порядок переноса Проекта на технические средства Заказчика

Все что не прописано и соответственно не утверждено Заказчиком в ТЗ Разработчик реализует по собственному усмотрению.

По результату составления ТЗ могут также измениться сроки реализации Проекта и его окончательная стоимость. Стоимость Проекта может отличаться от стоимости, предложенной изначально в коммерческом предложении и связано это с тем, что и Заказчик, и Разработчик смотрят на Проект одинаково (сквозь призму документа ТЗ).

3.5.6. Front-end

Основная задача этого этапа состоит в том, чтобы перевести разработанный и согласованный дизайн из графического формата (картинки) в статический HTML. Другими словами, на этом этапе происходит реализация концептов дизайна и подготовка материалов для интеграции шаблона в систему управления сайтом (контентом). Помимо статического интерфейса на этом этапе можно добавлять графические эффекты и реализовать интерактивность страниц, для которой не требуется взаимодействие с сервером. Необходимо в полной мере реализовать адаптивный дизайн или мобильную версию сайта. Максимально предусмотреть разные вариации расположения блоков и модулей, если это объявлено в ТЗ.

Необходимо прописать алгоритм выполнения данного этапа и систематизировать ключевые составляющие дизайна.

Рекомендуемая структура алгоритма:

- В зависимости от концепта проекта, составление плана верстки (тезисно).
- Разбивка дизайна на страницы.
- Выделение типовых модулей (формы, поля контента, дизайнерские (модульные) блоки).
- Выделение и систематизация служебных компонентов (шрифты, иконки, библиотеки, технологии, медиа контент).
- При необходимости вынести и прописать (псевдокодом) логически сложные элементы интерфейса или логику, которая реализована на стороне пользователя (фронте).

3.5.7. Back-end

На данном этапе происходит связка всех видимых объектов и форм Проекта:

- Создание административной панели под озвученную задачу Заказчика (например, если это e-commerce Проект, то настраивается панель добавления или управления товарным каталогом).

- Реализация (программирование) внутренней логики Проекта в соответствии с ТЗ.
- Внутреннее соединение и активизация функционала Проекта, соединение верстки (HTML) и программного кода.
- Интеграция с внешними учетными управленческими программами: CRM, ERP, биллинговые системы, ip телефония, прочее.

Важным на этом этапе является то, что в случае сложных интеграций и использования внешних данных, такие данные должны быть предоставлены Заказчиком строго в соответствии с форматом, описанным в ТЗ. Если на этом этапе такие данные не предоставлены или предоставляются Заказчиком в другом формате, то есть вероятность возникновения в будущем сложностей с корректной работой того или иного блока.

Результатом данного этапа является полностью работающий проект – весь функционал спроектирован, закодирован, отлажен и корректно выведен для визуального восприятия.

3.5.8. Тестирование, наполнение контентом.

Этот этап может частично идти параллельно с этапом программирования (или другими словами – к тестированию можно приступать в момент, когда этап программирования практически завершен). Основная задача – выполнить проверку соответствия полученного Продукта ТЗ и протестировать логику работы Проекта и отсутствие ошибок в функционале.

На этом этапе могут параллельно работать две команды или два специалиста: QA специалист и контент менеджер.

Контент менеджер обеспечивает наполнение корректным контентом (текст, картинки, видео). Это позволяет на раннем этапе учесть несоответствия или ошибки, которые могут в последующем иметь место.

При этом QA специалист составляет план тестирования и использует в своей работе соответствующие чек-листы:

- Чек-лист соответствия визуального восприятия и дизайна.
- Чек-лист корректной работы программной (внутренней) части Проекта.

Процесс тестирования подтверждает и фиксирует корректность:

- Функциональное поведение пользователя соответствует утвержденному дизайну и ТЗ
- Соответствие контента заявленному в дизайне и ТЗ. Особое внимание следует обратить на контент, который загружается автоматически при интеграции со сторонними сервисами и проверить его корректный вывод и формат.
- Отображение проекта на всех девайсах, предусмотренных для функционирования Проекта
- Отображение проекта во всех браузерах и операционных системах
- Соответствие параметров проекта утвержденным в ТЗ: скорость загрузки, масштабируемость, производительность.

Также на этом этапе составляется описание Проекта (Manual), который в момент сдачи Проекта передается Заказчику. Этот документ позволяет легко вникнуть в сам Проект

человеку, который не принимал активное участие в разработке и реализации Проекта (например, замена ответственного сотрудника у Заказчика)

Результатом этапа тестирования является баг репорт, который содержит перечень ошибок. Ошибки классифицируются следующим образом:

- критические функциональные ошибки, которые влияют на основной функционал и работоспособность сайта.
- некритичные или эстетические проблемы, несоответствие утвержденному дизайну. Должно быть устранены перед запуском.
- не важные. Улучшения или пожелания, которые могут быть или опущены, или скорректированы при наличии тайминга и бюджета.

3.5.9. Подготовка к запуску проекта.

На этом этапе команда Разработчика проходит по баг репорту и устраняет все ошибки согласно классификации. После устранения всех функциональных, визуальных, технических и контентных ошибок, проект переходит в стадию пользовательского тестирования.

Представители Заказчика, Разработчика, закрытая группа лиц тестируют проект, имитируя действия обычного пользователя.

Если в результате данного этапа появятся дополнительные пожелания к функционалу, дизайну проекта – эти доработки следует расценивать как дополнительную работу Разработчика, они требуют дополнительного времени на реализацию и дополнительного бюджета. Могут быть реализованы до запуска проекта с условием корректировки договора или составлением дополнительных приложений к нему или после запуска проекта как отдельный проект.

3.5.10. Перенос проекта на основной домен.

После того, как выполнены и закончены все предыдущие этапы и Проект готов к его запуску, происходит его перенос на основной хостинг и домен.

Организацией размещения проекта на хостинге занимается Разработчик. Хостинг по рекомендациям Разработчика обеспечивает Заказчик, делегируя ему необходимые для работы права. В случае нестандартных решений по хостингу или необходимости настройки сервера, этим процессом занимается представитель Разработчика если у Заказчика нет в штате соответствующего IT специалиста. Следует разделять работу Разработчика по разработке проекта и администрированию, и настройке хостинга. Как вариант – это могут быть разные компании.

Перед открытием проекта для использования пользователями, Разработчик должен проверить, что весь тестовый контент удален, очистить базы данных, удалить ссылки на сторонние приложения, проверить что проект открыт для индексации поисковыми системами если в исключительных случаях не требуется другого, установлены необходимые системы аналитики.

3.5.11. Промышленное тестирование Проекта.

Важным этапом в работе Проекта является его промышленное тестирование с рабочей нагрузкой с целью проверки устойчивой его работы. Поэтому на этом этапе желательно выполнить максимальный нагрузочный тест и проследить такие важные параметры, как:

- Скорость загрузки и обработки данных
- Пиковые нагрузки

Важно проверить одновременное многопользовательское администрирование интернет-проекта.

3.6. Сдача Проекта. Обучение персонала.

Перед тем, как подписать акт выполненных работ и официально передать Проект Заказчику, важно провести вводное обучение персонала, ответственного за поддержание/администрирование Проекта со стороны Заказчика и обсудить:

- Особенности административной панели в управлении Проектом
- Содержание Manual и как правильно его применять на практике
- Алгоритм взаимодействия в случае возникновения вопросов, связанных с управлением Проекта.

Как правило, оговаривается период поддержки со стороны Разработчика, на протяжении которого представитель Заказчика будет обращаться в установленном порядке для решения своих задач.

После завершения Проекта по требованию Заказчика Разработчик передает все исходные материалы, которые были выработаны в процессе разработки проекта: дизайны в исходных файлах в соответствующих программах, созданный контент, шрифты и прочее.

В случае если такой запрос не поступил Разработчик может передать материалы в течении гарантийного периода или в течении 90 дней после реализации проекта. По истечению данного срока или срока гарантии, Разработчик не обязан сохранять у себя материалы проекта.

Разработчик имеет право использовать проект в целях своего селфпромо: размещать в портфолио, подавать на конкурсы и фестивали, а также разместить на проекте ссылку на свою компанию с указанием статуса Разработчика. Это есть принятой нормой, но допускаются вариации по обоюдному согласованию сторон.

3.7. ГАРАНТИИ.

Гарантийные условия носят индивидуальный характер и зависят от сложности Проекта, внутренней политики Разработчика и направлены на урегулирование отношений процессуально-юридического характера.

Гарантийные условия обязательно согласовываются и прописываются в Договоре о разработке проекта. Рекомендуемый период гарантийных обязательств составляет от 90 до 180 календарных дней в зависимости от типа проекта и по согласованию сторон.

Цель гарантийного периода заключается в исправлении ошибок или неточностей, которые не были обнаружены в период тестирования и после запуска проекта.

Исправлению за счет Разработчика подлежат только неисправности, которые регламентированы в ТЗ и не выходят за пределы утвержденного функционала.

Гарантийные обязательства не распространяются на проект в случае, если ошибки в работе возникли в результате изменения в каких-либо сторонних платформах, браузере, устройствах, операционных системах или других сторонних системах, не зависящих от Разработчика.

Гарантийные обязательства аннулируются на любом этапе гарантийного периода в случае если Заказчик предоставил доступ к коду и административной панели проекта третьим лицам.

После окончания гарантийного периода все пожелания, возникающие ошибки, необходимые доработки будут рассматриваться Разработчиком как новые задачи в индивидуальном порядке.

Во время гарантийного периода и после него, при условии, что перед запуском проекта, Разработчик провел обучение уполномоченным представителям Заказчика, Разработчик не обязан проводить консультации и обучения в качестве гарантийных обязательств. Условия консультаций и обучения дополнительного персонала Заказчика устанавливаются Разработчиком в индивидуальном порядке.

3.8. Изменения проекта после запуска.

Разработчик оставляет за собой право взимать плату за любую работу и задачи, возникающие после запуска проекта, не выявленные на этапе тестирования.

Все необходимые изменения проекта должны быть предоставлены Заказчиком в письменном виде для оценки Разработчиком требуемого срока реализации и бюджета. Разработчик берет новые задачи в работу после утверждения Заказчиком, но согласно своего внутреннего графика работ, ориентированного на загрузку команд.

4. Требования к информации, предоставляемой Заказчиком.

Информация, которая предоставляется Заказчиком, должна соответствовать определенным стандартам/расширениям и не нарушать авторские права.

Тип информации	Применяемый формат	Уточнения
Картинки	.JPG; .PNG; .GIF; .JPEG	.png; .gif – данные форматы не оптимизируются
Векторная графика	.SVG; .EPS .AI	.eps; .ai – исходные файлы, которые можно использовать
Видео	.MP4; .WEBM; .OOG	.webm; .oog – нужны для кросс-платформ. Min от Заказчика нужна информация в 2-х форматах
Аудио	.MP3; .WAV	
Шрифты	.TTF; .OTF	Если у Заказчика есть корпоративный шрифт, то он должен его предоставить с учетом кириллицы и латиницы
Базы данных	.XLS .XML; .XLXS;	

5. Персонал, принимающий участие в процессе разработки интернет-проекта, и его основной функционал.

Важно понимать, что за успешную реализацию Проекта отвечает команда, которая состоит как из представителей Разработчика, так и со стороны Заказчика.

Со стороны Разработчика может быть следующий состав команды Проекта:

Позиция	Основной функционал
Analyst	Анализ сегмента и ниши рынка, ЦА, позиционирование Продукта
Art Director	Креатив и общая концепция дизайна Проекта
Back-End Developer	Программирование и внутренняя логика Проекта, системы управления, работа с фреймворками, плагинами, выгрузками.
Copywriter	Создание уникального контента Проекта: слоганы, заголовки, статьи, описание Товаров и Услуг.
Content manager	Наполнение Проекта текстовой и графической информацией, предварительная подготовка информации.
Designer	Разработка дизайна Проекта и графических материалов.
Front-End Developer	Верстка шаблонов сайта и создание пользовательского интерфейса
Project manager	Общее управление Проектом и ходом его реализации. Отвечает за качество и сроки реализации Проекта, бюджет и KPI.
QA specialist	Тестирование функционала, дизайна и наполнения Проекта с учетом заданных параметров, проведение нагрузочного тестирования
Strategist	Сегментирование пользователей, поведенческие характеристики, стратегия Проекта и KPI
Team Lead	Оптимизация и эффективное использование ресурса в технической команде, решение нестандартных задач, общий технический аудит
UI/UX Designer	Анализ поведения Пользователя и разработка пользовательских интерфейсов и построения Customer Journey

6. Технические требования к разрабатываемому проекту

К техническим требованиям относятся:

● Разрешения экрана и масштаб

При разработке дизайна сайта необходимо предусматривать и фиксировать это в ТЗ - как ведет себя сайт при отображении на мониторах с разным разрешением экрана. При этом следует учесть минимальное разрешение (например, 720 px), при котором дизайн сайта перестраивается под планшеты или смартфоны по технологии адаптивности.

С другой стороны, также необходимо учесть максимальное разрешение экрана, на которое рассчитывается сайт. В некоторых случаях необходимо представить отдельный вариант дизайна для мониторов с разрешением более 2500px.

Такой же подход важен и при масштабировании изображения экрана. В ТЗ отдельно прописывается как ведет себя сайт при зумировании экрана как в сторону увеличения, так и уменьшения масштаба.

● Отображение на устройствах

С учетом последних требований поисковой системы Google к ранжированию в выдаче страниц и их ориентиром на Mobile Friendly, особое внимание необходимо уделять адаптивности сайта и его удобному и целостному отображению на устройствах всех типов.

Существует два подхода к визуальному отображению сайтов на устройствах различных типов: десктопная и мобильная версия сайта и адаптивный дизайн.

В зависимости от сущности проекта, его структуры и задач, выбирается один из подходов: создание мобильной версии в дополнение к десктопной или адаптивный дизайн.

Минимальное разрешение на которое следует ориентироваться в разработке составляет 320 px, максимальное типовое разрешение – 1920 px. ТЗ является регламентирующим документом, определяющим отображения на смартфонах, планшетах и десктопах. Отдельно необходимо учесть дизайн и верстку с учетом горизонтальной и вертикальной ориентации на смартфонах и планшетах.

Работоспособность

● Скорость загрузки

Скорость загрузки сайта относится к одному из параметров качества функционирования сайта и существенно влияет на общую работоспособность сайта. При отсутствии дополнительных требований или ограничений необходимо стремиться к максимальной скорости загрузки и минимальному времени ожидания.

При этом общую скорость работы сайта можно разделить на два типа:

- скорость загрузки статических страниц
- скорость обработки и отдачи данных (фильтры, сортировки)

Для измерения скорости загрузки сайта существуют специальные инструменты и сервисы:

Основными источниками нагрузки и долгой загрузки могут быть: тяжелый видео контент, не оптимизированные изображения и другой контент, который физически находится на внешних ресурсах и передается для загрузки в проект, а также элементы кода, такие как java scripts. При обнаружении проблем со скоростью загрузки и определения зоны, в которой эти проблемы лежат требуется оптимизация изображений и кода и проверка работоспособности и корректности внешних ссылок, по которым загружается контент на проект. Также проблемы с нагрузками могут лежать в зоне характеристики хостинга и настройки серверов. Потому этим параметрам необходимо уделить отдельное внимание.

- Пиковая нагрузка

Перед запуском проекта необходимо протестировать его нагрузочные характеристики. Нагрузка - это параметр, который показывает объем требуемого ресурса необходимого для обеспечения нормальной работоспособности проекта.

Хостинг и сервера

При разработке проекта необходимо учитывать технические характеристики сервера, на котором будет располагаться проект.

Существует два основных подхода к физическому размещению проекта:

- Размещение проекта на хостинге
- Размещение проекта на сервере: VPS или выделенный сервер

Браузеры

При разработке проекта необходимо понимать и учитывать специфику визуального отображения и поддержания тех или иных технологий браузерами. Рекомендовано при проектировании проекта и понимания целевой аудитории проекта определить самые популярные/востребованные ЦА браузеры и ориентироваться в разработке на корректное отображение именно в этих браузерах.

Необходимо учитывать, что некоторые браузеры или отдельные версии не поддерживают многих популярных технологий верстки, motion design и интерактивных спецэффектов. В таком случае либо разрабатывается версия без учета этих технологий, либо происходит отказ от этих технологий. В любом случае, это должен быть подконтрольный процесс. Минимальные технологии, на которые необходимо ориентироваться в разработке проектов и которые должны поддерживаться браузерами - HTML5, CSS3, ECMAScript 6+

Безопасность

Безопасность проекта глобально состоит из 3-х составляющих: безопасность программной части, безопасность сервера, безопасность при администрировании проекта (человеческий фактор).

Безопасность программной части проекта включает в себя требования к коду, скриптам, системе управления контентом, которые минимизируют риск возникновения уязвимостей, позволяющих злоумышленнику получить доступ к базе данных, файловой системе или панели администратора сайта.

При использовании популярных готовых систем управления сайтом, необходимо постоянно следить за обновлениями и патчами и оперативно обновлять CMS до последней доступной версии.

При работе с индивидуально написанной системой, на основе фреймворка, как правило обеспечивается более высокая степень защищенности проекта. Для обеспечения еще более высокой степени защиты необходимо выполнять сканирование проекта на наличие уязвимостей и проверять исходный код средствами статического анализа исходного кода (RIPS). Делать это необходимо как при сдаче готового проекта, так и через определенный установленный период технического обслуживания.

Вторым важным моментом, влияющим на безопасность сайта в целом, является **хостинг**, на котором размещается сайт. Хостинг может быть shared («общий») или dedicated («выделенный»).

Для shared-хостингов ответственность за безопасную настройку сервера лежит на администраторе хостинг-компании. Для dedicated-сервера (VDS/VPS/DDS) эта ответственность лежит на владельце сервера.

В зависимости от типа интернет-проекта, объемов ожидаемой нагрузки и необходимой степени безопасности руководитель проекта должен подобрать наиболее подходящий пакет хостинга - стандартный хостинг, выделенный (VDS/VPS/DDS) сервер или облачные решения для обеспечения максимальной безопасности.

Настройку выделенных серверов и серверов с использованием облачных технологий должен проводить опытный системный администратор, который изолирует сайт от остальной части системы, максимально ограничит свободу скриптов и область их видимости, а также организует механизмы контроля целостности файловой системы, систему резервного копирования и логгирования.

Третий момент – это администрирование сайта. И для обеспечения безопасности необходимо выполнять несколько простых требований:

- Правильно распределять роли доступа к администрированию и функционалу сайта
- Не пренебрегать сложными паролями
- Не хранить пароли в почте, на ftp

Прочие рекомендации для обеспечения безопасности проекта:

- Протокол HTTPS – использование безопасного протокола с шифрованием данных для защиты данных пользователей, которые они оставляют на сайте
- Защита от спама. Защита форм от спамовских рассылок
- Защита от вирусов
- Защита от брутфорса, защита заголовков, защита от csrf.
- Дополнительно: перехват и валидация запросов, кеширование базы данных, валидация IP-адресов по геолокации, валидация заголовков, серверный рендеринг фронтэнда, использование готовых решений для облачных хостингов

7. Технологии и инструменты, применяемые при создании интернет-проектов.

Технология	Описание технологии
AMP	Accelerated Mobile Page – технология верстки
ECMAScript 6,7,8	JavaScript Language. Скриптовый язык, для реализации логики сайта на стороне клиента. Версии от 2015 года, обеспечивает хорошую скорость работы и доступ к современным АПИ браузеров и устройств.
ECMAScript 5.1	JavaScript Language. Скриптовый язык, для реализации логики сайта на стороне клиента. Версию 5.1 стоит использовать для поддержки IE-11, Safari 10.1.
jQuery 3.*	Библиотека функций. Набор готовых решений написанных на JavaScript.
jQueryUI	Библиотека функций для кастомизации пользовательского интерфейса.
HTML 5	Язык гипертекстовой разметки. Актуальная версия стандарта 5.2.
CSS - CSS3	Каскадные таблицы стилей.
SASS, LESS	Абстракция каскадных таблиц стилей.
PWA	Progressive Web App. Гибрид мобильного приложения и сайта.
WebGL, Canvas	API для создания графики на веб-страницах
GSAP, PIXI.js	Инструменты для работы с анимацией на веб страницах
Apache	Свободный веб-сервер
nginx	Свободный веб-сервер
PHP	Hypertext Preprocessor. Актуальная версия 7.2.5
Laravel	framework based on PHP ver. 7.1. Support PHP ver. 7.2 (for Argon)
Vue.js	JavaScript Framework
Tailwind css	Модульный CSS Framework
SQLite	Очень мощная встраиваемая СУБД
MySQL	Самая популярная и распространённая СУБД
PostgreSQL	Наиболее продвинутая СУБД
Nuxt.js	Server-side render for Vue.js (если такое надо:))
Memcached	Сервис кэширования данных в оперативной памяти на основе хеш-таблицы

Redis	Сетевое журналируемое хранилище данных типа «ключ — значение», нереляционная высокопроизводительная СУБД
-------	----------------------------------------------------------------------------------------------------------

ПРИЛОЖЕНИЕ 1

Краткая процедура разработки интернет проекта для представителей компании Разработчика (Агентства)