



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ  
УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

---

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И  
ТЕХНОЛОГИЧЕСКОГО ОБРАЗОВАНИЯ**  
**Кафедра информационных технологий и электронного обучения**

Основная профессиональная образовательная программа  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Направленность (профиль) «Технологии разработки программного обеспечения»  
форма обучения – очная

### **Курсовая работа**

« Разработка обучающего информационного ресурса "Инструменты для  
управления проектами »

Обучающегося 3 курса  
Чалапко Егора Витальевича

Научный руководитель:  
Кандидат физико-математических наук,  
доцент кафедры ИТиЭО  
Жуков Николай Николаевич

Санкт-Петербург  
2021

## ОГЛАВЛЕНИЕ

ВСТУПЛЕНИЕ .....	3
1. Аналитическая часть .....	4
1.1 Методологии разработки сайта.....	4
1.2 Возможные методологии разработки.....	4
1.3 Выбранная методология .....	6
1.4 Техническое задание .....	8
2. Практическая часть .....	11
2.1 Дизайн .....	11
2.2 Содержание сайта.....	18
2.3 Способы создания веб-ресурса .....	26
ЗАКЛЮЧЕНИЕ .....	40
СПИСОК ЛИТЕРАТУРЫ.....	41

## ВСТУПЛЕНИЕ

Управление проектами всегда являлось важной частью проведения и успеха поставленного проекта. Из-за этого, и по многим другим причинам, управление проектами получило своё развитие не только как предмет, но и как профессиональное направление. Подобное развитие потребовало создание различных методологий и правил, которые могут облегчить как управление рабочим процессом, так и сам рабочий процесс.

Но, как известно, «прогресс не стоит на месте» и всё продолжает развиваться и двигаться вперёд. Технологии, программы и предложения продолжают внедряться во все сферы обычной жизни. Этот прогресс позволяет упрощать, ускорять и улучшать людской труд, давая им инструменты для работы. Такие инструменты уже существуют и для управления проектами.

В данной курсовой работе будет затронута одна из этих систем и принципы работы с ней. Задача этой курсовой работы – разработка обучающего информационного ресурса, позволяющего пользователям узнать чуть больше о выбранной системе, изучение принципов работы с ней и способов облегчения управления проектами, и, что немаловажно, анализ и описание методологий и инструментов, используемых для разработки данного информационного ресурса.

Цели курсовой работы:

1. Провести управление разработкой интернет-ресурса;
2. Разработать информационный интернет-ресурс по теме "инструменты управления программными проектами".

Задачи курсовой работы:

1. Изучить методологии разработки программного обеспечения и выбрать подходящую;
2. Исследовать способы создания веб-базируемого ресурса;
3. Рассмотреть различные инструменты управления проектами;
4. Разработать интернет-ресурс по одному из рассмотренных инструментов управления проектами.

Объект исследования:

Использование инструмента управления проектом.

Предмет исследования:

Использование веб-базируемого ресурса для изучения инструментов управления проектами.

## 1. Аналитическая часть

### 1.1 Методологии разработки сайта

Для работы над проектом необходимо сначала выбрать методологию разработки приложения. Методология разработки ПО – это система, определяющая порядок выполнения задач, методы оценки и контроля. Модели разработки ПО выбирают, исходя из направления проекта, его бюджета, сроков реализации конечного продукта, а также внимание стоит обратить и на характер и темперамент руководителя проекта и его команды. Подходы разработки ПО отличаются друг от друга тем, как этапы жизненного цикла программного обеспечения взаимосвязаны между собой внутри цикла разработки.

Методология разработки ПО позволит лучше отслеживать стадии жизненного цикла приложения, и что необходимо сделать в каждой из них, а также позволит сделать разработку более эффективной.

### 1.2 Возможные методологии разработки

В данной главе будут разобраны возможные методологии разработки данного проекта.

#### «Водопад» или каскадная модель (Waterfall Model)

Классическая методология, используемая «с незапамятных времен». Представляет собой строго последовательное выполнение всех стадий разработки. Иными словами, новая стадия не начинается до тех пор, пока не будет полностью закончена предыдущая.

Каскадная модель очень удобна для управления проектом, поскольку процесс разработки легко отслеживается. Это дает возможность жесткого контроля над процессом разработки, что позволяет достаточно точно заранее определить сроки окончания и общую стоимость проекта.

Однако такая жесткость имеет и негативную сторону. «Водопад» хорошо подходит для проектов с предельно четкими требованиями и заранее продуманными способами реализации. Но если в техническом задании есть «туманные» моменты, которые можно трактовать двусмысленно, каскадная методология становится крайне неудобной. «Водопад» не предусматривает возможности откатить разработку на одну-две стадии назад, отсутствует возможность протестировать отдельный аспект до полного окончания разработки. По этой причине невозможно вносить изменения, поправки и корректировки в уже сделанную часть работы, либо внесение таких поправок резко повышает стоимость проекта.

Таким образом, каскадная методология подходит исключительно для тех проектов, где требования в техническом задании предельно точны, понятны и зафиксированы на

бумаге, а какие-либо разночтения или недопонимания отсутствуют. Также очень желательно использовать данную методологию только для относительно небольших проектов.

### V-образная модель (V-Model)

Данная модель имеет в целом те же принципы последовательной «шаг-за-шагом» разработки, что и каскадная, но отличается от нее одним принципиальным моментом — на каждом этапе осуществляется тестирование готовой части проекта.

V-образную модель обычно используют при разработке программного обеспечения, предназначенного для важных систем, где недопустимы перебои в их работе. К примеру, при создании программного обеспечения для мониторингового медицинского оборудования, различных систем безопасности и т.д. Словом, везде, где ошибки и недочеты в программном продукте могут иметь серьезные последствия.

В целом можно уверенно говорить о предпочтительности V-образной модели для тех проектов, которые требуют тщательного тестирования всех аспектов от удобства интерфейса до системной стабильности и отсутствия узвимоостей для внешнего вмешательства.

### Инкрементная модель (Incremental Model)

Данная методология используется для проектов, предусматривающих несколько вариантов (сборок) готового продукта. Зачастую разработка ведется несколькими циклами, то есть в итоге получается своего рода «мульти-водопад». При этом в каждом цикле имеются свои этапы и создаваемые модули. Для каждого модуля предусмотрены собственные этапы уточнения требований, создания проекта, кодирования, тестирования и т.д.

Инкрементная модель предполагает особую последовательность созданияборок: сначала реализуется основной проект (базовая сборка), затем на ее основе создаются новые сборки с новыми функциями, называемыми «инкрементами».

Разработка по инкрементной модели хороша для тех проектов, в которых четки и ясны не только базовые требования к системе, но и запросы на внесение изменений тоже ясны, а сами изменения легко реализуются. При этом вполне допускается, что отдельные функции и новые сборки могут дорабатываться уже после внедрения на практике базовой сборки.

### Быстрая разработка приложений или «RAD Model»

Представляет собой разновидность описанной выше инкрементной модели. Ключевым отличием является то, что компоненты проекта (модули) или разные сборки разрабатываются не поочередно одной командой, а параллельно несколькими командами. В условиях жестко лимитированного времени созданные одновременно модули собирают в единый рабочий прототип. В итоге удастся предоставить заказчику рабочую систему в предельно сжатые сроки.

Важным условием применения данной методологии является наличие нескольких высококвалифицированных команд специалистов. Следствием такого подхода становятся высокие расходы на оплату услуг большого количества задействованных спецов и рабочих инструментов, которые они используют.

### Гибкая модель разработки (Agile Model)

Ключевая особенность данной методологии заключается в максимальной прозрачности процесса разработки для заказчика, у которого есть возможность отслеживать буквально каждую итерацию и одобрять ее либо требовать переделки. Таким образом, полностью исключается малейшая вероятность сделать совсем не то, чего хотел клиент.

Очевидным недостатком гибкой модели является сложность предварительной оценки трудозатрат и стоимости проекта. Однако в условиях отсутствия четких требований и невнятного ТЗ, когда заказчик сам весьма смутно понимает, что ему нужно, гибкая модель является единственно возможной для использования.

Важным атрибутом гибкой методологии является проведение непродолжительных ежедневных встреч, которые именуются «Scrum», а также регулярных собраний раз в неделю или реже, именуемых «Sprint».

Методология хорошо себя показывает при разработке больших проектов, либо проектов, которые нужно постоянно адаптировать к меняющимся условиям рынка.

### 1.3 Выбранная методология

Из представленных методологий, более всего данному проекту подойдет Agile model, так как она не предполагает четких начальных требований к проекту и подходит небольшим командам. Помимо этого, данная методология позволяет вносить изменения на любых этапах проекта. Всё это позволит нам, имея примерные представления о конечном продукте, вести работу над ним, внося необходимые правки по ходу разработки.

Конкретно будет использоваться подтип методологии разработки Agile, называемый Kanban.

Kanban — это метод улучшения процессов разработки и часть agile-философии. Цель Kanban — получать готовый качественный продукт вовремя.

Kanban начинается с визуализации, чтобы процессы были на виду у команды. Для этого используют специальную доску и набор карточек или стикеров.

Доска — это обязательный элемент для гибкой методологии. Она есть в Scrum, есть и в Kanban. Каждый член команды получает к ней доступ в любое время и видит, на каком этапе находится задача. Kanban-доска подстраивается под любой процесс и применяется в любой области.

У каждого проекта есть план процесса работ. Сначала производится его анализ и разделение доски на столбцы, которые отражают этапы. Имена столбцов меняются в зависимости от проекта, но важно сохранять их последовательность — это ключевая ценность Kanban, которую называют потоком. Kanban-карточки — это задачи, которые движутся по потоку и перетекают в другие столбцы в зависимости от их состояния. На карточке или стикере пишут название задачи и прикрепляют в начало доски. С помощью kanban-доски легко вести несколько проектов одновременно, используя карточки разных цветов: один цвет — один проект.

На доске отражаются все процессы. Команда их анализирует и устраняет слабые места. В Kanban это называется управлением потоком. Команда в Kanban — единый механизм. Если кто-то не справляется, то страдает общее дело. Работу планируют на доске, поэтому каждый может увидеть свой вклад и ценность для проекта.

В Kanban смешались принципы agile-методологий и lean-мышления. Здесь нет жёстких правил, но есть принципы, на которые можно опираться.

Основные принципы KANBAN:

- Уважать и использовать то, что есть сейчас: имеющиеся роли, обязанности и должностные инструкции.
- Постоянно оптимизировать и совершенствовать процесс разработки, но не допускать слишком резких перемен.

Визуализация помогает видеть картину целиком и корректировать отдельные её части, понимая, как изменения затронут весь проект. Получить результат точно в срок возможно, если контролировать нагрузку команды.

Важно найти баланс: выбрать темп работы, который удобен команде и не вредит срокам проекта. Для этого в Kanban учитывают время выполнения каждой задачи. Так команда понимает, что занимает больше времени, а что — меньше, и может правильно организовать работу.

Kanban был выбран для разработки не только потому, что он подходит для небольших команд, но и потому, что он позволяет максимально удобно выстроить сроки на выполнение определённых задач, а также помогает отлично визуализировать ход выполнения работы.

## 1.4 Техническое задание

В этой главе будет описано техническое задание на данный проект.

### 1. ВВЕДЕНИЕ

В данном разделе содержится техническое задание для разработки информационного интернет-ресурса, посвященного инструменту управления проектами. Ресурс ориентирован на любых посетителей, заинтересованных в управлении проектами или в теме упп и инструментов упп.

### 2. НАЗНАЧЕНИЕ

Основная задача интернет-ресурса — информирование посетителей о том, что такое управление проектами, и подробный разбор одного из инструментов управления проектами.

### 3. СТРУКТУРА И ОПИСАНИЕ

На сайте будет 4-5 разделов:

- Главная страница;
- Страница автора;
- Страница с дополнительной информацией;
- Страница с основным содержанием сайта;
- (Возможная страница с инструкцией по использованию инструмента).

### 4. РАЗДЕЛЫ И СТРАНИЦЫ

#### 4.1 Главная страница:

Первая страница, на которую попадает пользователь, содержит краткую информацию об управлении программными проектами, о том, что такое инструменты управления программными проектами. Содержит также информацию о самом сайте, для чего он был создан, и что на нём можно найти.

#### 4.2 Страница Автора:

На этой странице будет изложена краткая информация об авторе сайта. Также на этой странице будет размещена контактная информация, которая поможет связаться с автором при помощи разных ресурсов, если потребуется внести какие-то изменения в содержание сайта.



#### 4.3 Страница с дополнительными ресурсами:

Раздел будет содержать ссылки на ресурсы по теме. Ссылки будут направлять на сайты с источниками информации, использованной на ресурсе и дополнительной информацией по теме (сайты, видеоматериалы, статьи, блоги и т.п.)

#### 4.5 Страницы с основным содержанием сайта:

На них будет приведена информация о выбранном инструменте управления проектами, возможно история разработки и применение другими компаниями. Структуру можно продумать следующим образом: будет иметься одна страница, на которой будет приведена информация о самом инструменте, его истории и т. д. Затем с этой страницы можно сделать переход на следующую, с подробным разбором возможностей данного приложения и возможностью их применения.

### 5. ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ

Перечень технических требований:

- Корректное отображение в Google Chrome, Safari;
- Оптимизированная мобильная версия;

### 6. СЦЕНАРИИ

Сайт будет использоваться:

- Для краткого изучения темы управления проектами и инструментов упп;
- Для изучения указанного инструмента управления проектами;
- Для изучения возможностей данного инструмента, способов его использования.

### 7. КОНТЕНТ

Содержание сайта будет составлено в соответствии с указаниями выше. Основным инструментом упп, который будет исследован, был выбран Trello.

### 8. УСЛОВИЯ

Общий срок создания ресурса составляет 51 день. Из них:

- 9 — выбор способа создания сайта и создание макетов;
- 21 — дизайн сайта, сбор и форматирование содержимого;
- 21 — разработка и деплой сайта.

#### 8.1 Этапы создания сайта:

1. Выбор способа создания сайта. Существуют различные способы создания веб-ресурса. Будь то создание сайта при помощи соответствующих программ или приложений

для сборки сайта, аналогичных приложений для сборки блога, или же собственноручное написание кода и дизайна для сайта, с последующим деплоем через какой-либо сервис. Данный выбор невероятно важен, так как он не только по-своему повлияет на процесс работы над информационным ресурсом, но также повлияет и на вид самого ресурса.

2. Разработка макета сайта. Создание плана того, как должен строиться и функционировать ресурс, какие на нём должны быть разделы и страницы, примерное представление того, как это можно организовать имеющимися средствами. Возможное создание набросков.

3. Дизайн сайта. Выбор цветовой палитры и примерного визуального содержания и представления информационного ресурса. Представление того, как сайт может выглядеть, и как подобные идеи можно организовать. Возможное создание макетов дизайна.

4. Сбор материалов для заполнения сайта. Поиск содержимого сайта. Поиск необходимой информации, сбор скриншотов, материалов, ссылок на электронные ресурсы и организация полученной информации. Представление того, как эта информация будет выглядеть при размещении на электронный ресурс.

5. Разработка и сборка веб-ресурса. Будь то написание кода, сборка и организация сайта на каком-то ресурсе, программирование и применение дизайна, размещение информации и проверка точности работы сайта.

6. Деплой. Возможный деплой сайта на какой-то сервис, или выпуск сайта в общий доступ.

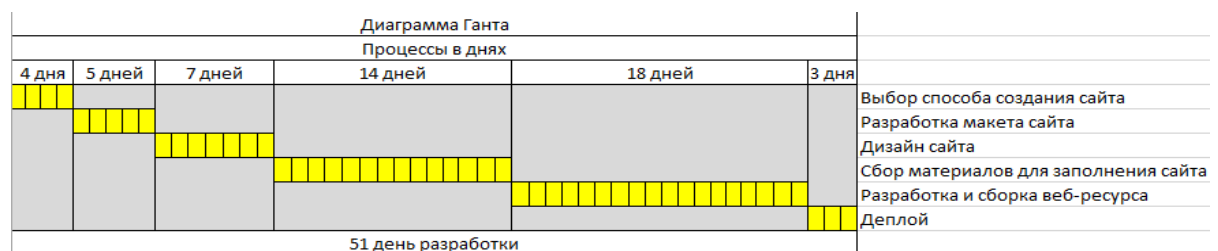


Рисунок 1. Диаграмма Ганта для разработки сайта

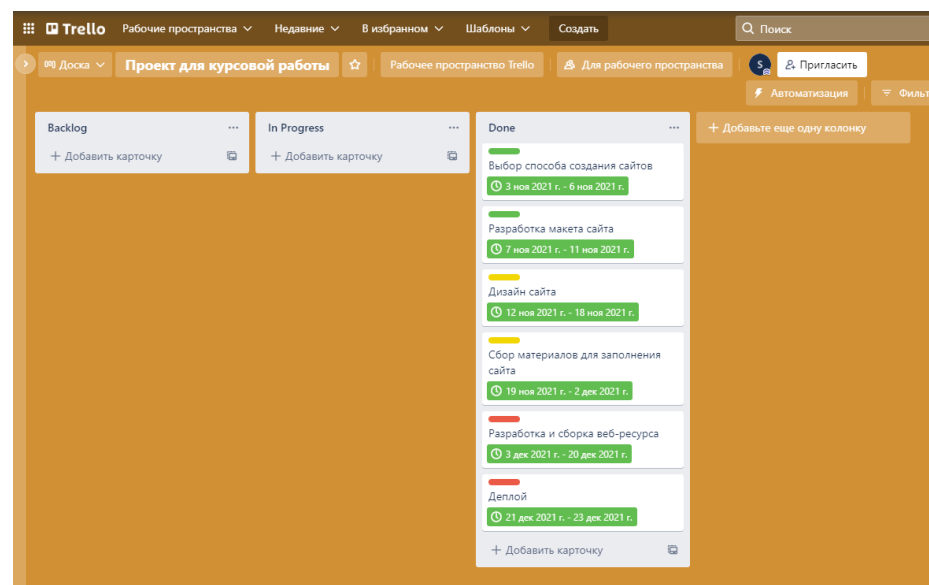


Рисунок 2. Канбан доска проекта

## 2. Практическая часть

### 2.1 Дизайн

В этой главе будет подробно разобран возможный вид сайта с различными примерами визуального оформления. По большей части, дизайн сайта будет зависеть от способа создания сайта. Рассмотрим несколько дизайнов в зависимости от выбранного метода.

#### Блог при помощи сервиса Blogger

Blogger — веб-сервис для ведения блогов, с помощью которого любой пользователь может завести свой блог, не прибегая к программированию и не заботясь об установке и настройке программного обеспечения.

Blogger – невероятно удобная система, которая позволяет создать собственный блог. С ней очень просто работать, так как она предоставляет набор заготовленных дизайнов и шаблонов, имеет понятную модульную систему, которая позволяет настроить систему по своему желанию.

Рассмотрим несколько предложенных шаблонов для блога:

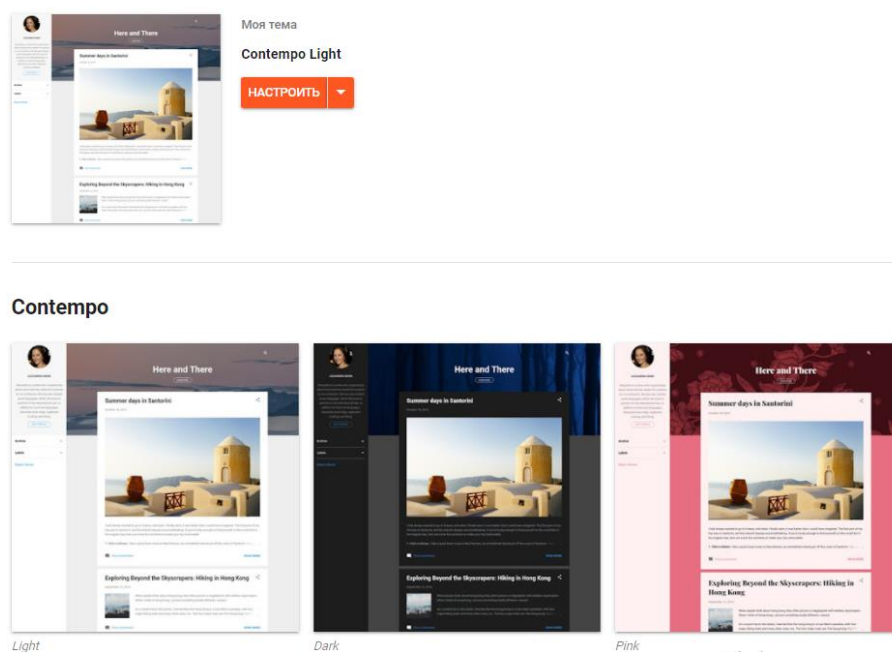


Рисунок 3. Шаблоны Blogger (1)

Simple

ещё

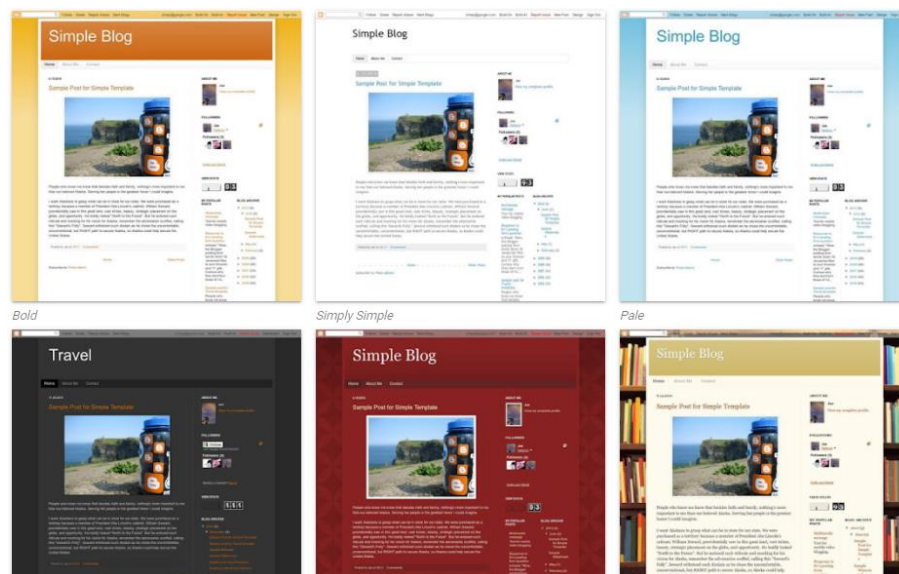
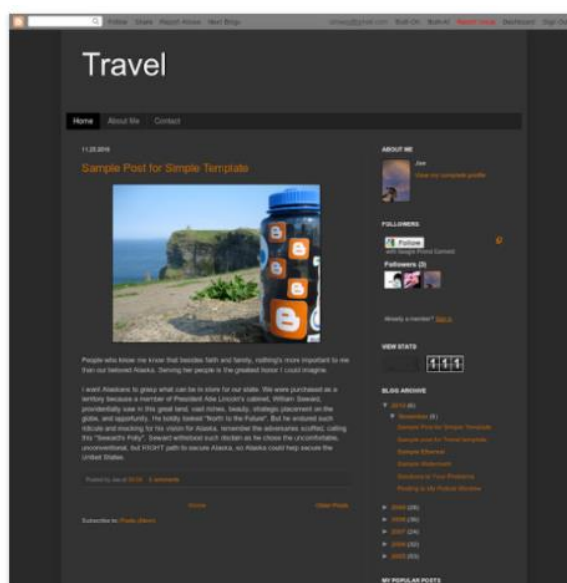


Рисунок 4. Шаблоны Blogger (2)

Из предложенных заготовленных вариантов больше всего нашему проекту подходят варианты «Simple Dark» и «Notable Coral»:



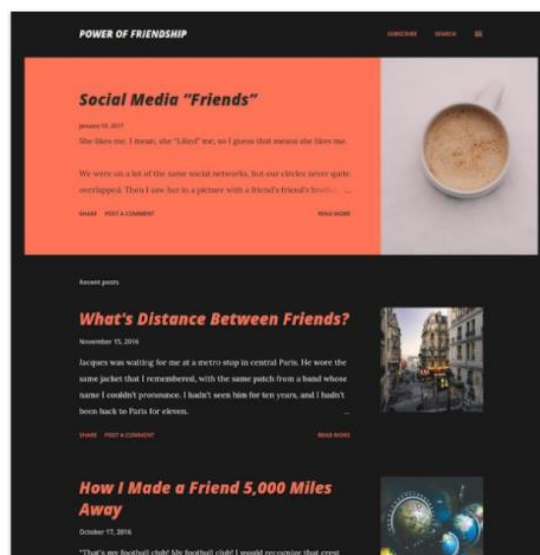
Simple Dark

ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР

НАСТРОИТЬ

ПРИМЕНИТЬ

Рисунок 5. Шаблон Simple Dark



Notable Coral

ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР

НАСТРОИТЬ

ПРИМЕНИТЬ

Рисунок 6. Шаблон Notable Coral

Эти варианты обладают простыми, спокойными и приятными дизайнами, которые не будут отвлекать посетителя страницы от основного материала, при этом не являются броскими или наоборот скучными, что позволит расширить аудиторию посетителей.

Сайт, при помощи платформы «Google Sites»

Сайты Google — упрощённый бесплатный хостинг. Позволяет при помощи технологии wiki сделать информацию доступной для людей, которые нуждаются в её быстрой подаче. Пользователи сайта могут работать вместе, добавлять информацию из других приложений Google, например, Документы Google, Календарь Google, YouTube, Google Фото и из других источников.

Google Sites – довольно удобный, хоть и очень простой и ограниченный конструктор сайтов. Предоставляет хоть и небольшой, но достаточный набор функций и шаблонов для создания собственного веб-ресурса.

Рассмотрим шаблоны, которые предоставляет данный сервис:

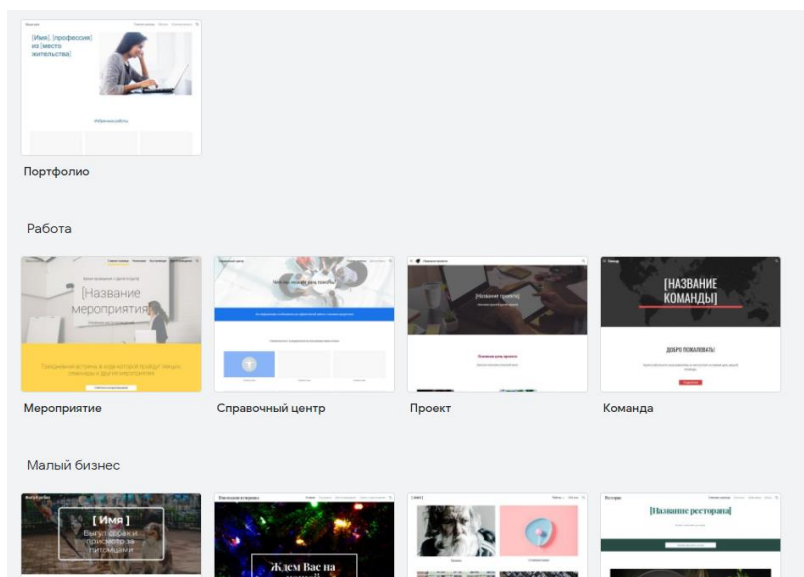


Рисунок 7. Шаблоны Google Sites

Данный сервис не предоставляет большого количества шаблонов, но данные шаблоны обладают неплохими возможностями по настройке. Из наиболее подходящих хочется отметить следующие шаблоны:

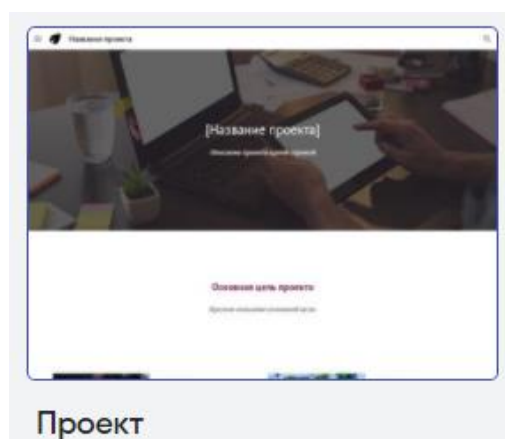


Рисунок 8. Шаблон «Проект»

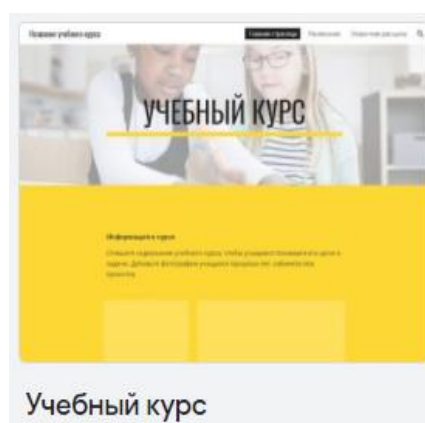


Рисунок 9. Шаблон «Учебный курс»

Данные шаблоны обладают простыми и спокойными дизайнами, которые могут подойти этому проекту. Несмотря на то, что их темы и цветовая палитра могут не совпадать с желаемым результатом нашей работы, благодаря возможностям ресурса, их можно настроить под нужды проекта.

Сайт, написанный вручную на html

HTML — стандартизированный язык разметки документов для просмотра веб-страниц в браузере. Веб-браузеры получают HTML документ от сервера по протоколам HTTP/HTTPS или открывают с локального диска, далее интерпретируют код в интерфейс, который будет отображаться на экране монитора.

Элементы HTML являются строительными блоками HTML страниц. С помощью HTML разные конструкции, изображения и другие объекты такие как интерактивная веб-формы могут быть встроены в отображаемую страницу. HTML предоставляет средства для создания заголовков, абзацев, списков, ссылок, цитат и других элементов.

Когда дело доходит до написания собственных сайтов с нуля – это, конечно, html. Хотя и написание на html является сложным и время затратным процессом, так как вся структура сайта пишется и программируется с нуля, требуя затем ещё и публикации, этот процесс также является одним из самых дружелюбных в плане дизайна. Ведь, работая с нуля над сайтом, также начинается работа с нуля и над дизайном. Но это позволяет максимально подстроить дизайн под свои нужды, позволяет воплотить в жизнь все задумки визуальной части, на сколько позволит отведённое время.

Для упрощения и ускорения работы рассмотрим шаблоны уже существующих сайтов, на которые мы сможем ориентироваться во время работы над сайтом проекта:

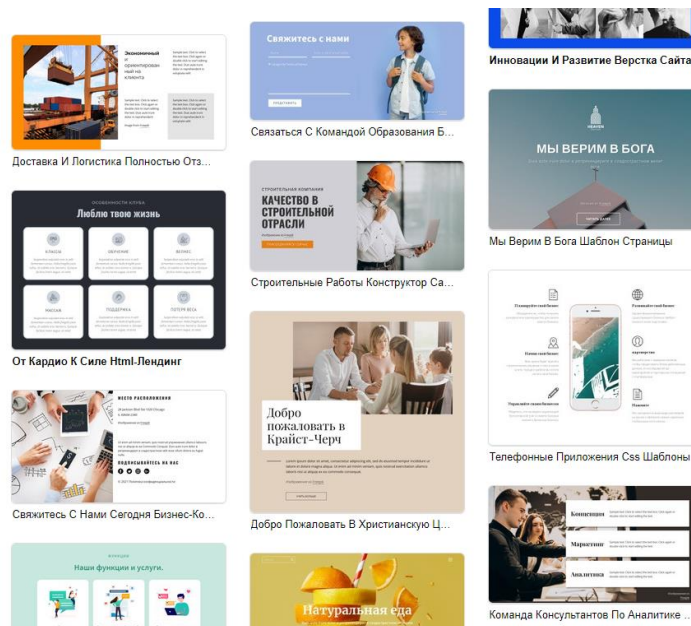


Рисунок 10. Шаблоны html (1)

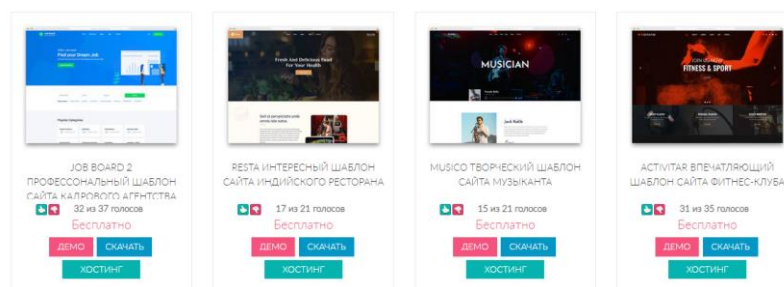


Рисунок 11. Шаблоны html (2)

Из исследованных шаблонов для нашей цели больше всего подойдут следующие:



Рисунок 12. Шаблон ARIALOGIC





Рисунок 13. Шаблон html

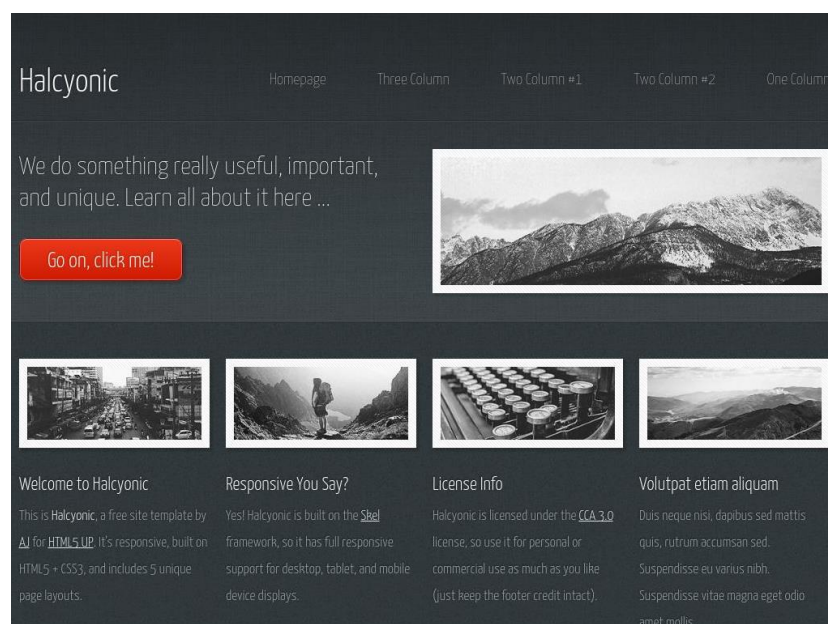


Рисунок 14. Шаблон Halcyonic

Данные шаблоны отличаются простотой дизайнов и спокойными приятными цветами. На основе данных шаблонов, используя возможности html, можно будет создать отличный сайт, который будет приятен для посещения любым пользователям, заинтересованным в теме инструментов управления проектами.

Рассмотрев, изучив и выбрав интересующий нас дизайн сайта, можно переходить к поиску информации для наполнения данного электронного ресурса.

## 2.2 Содержание сайта

В этой главе будет представлен примерный вариант отформатированной информации, собранной для наполнения веб ресурса. Здесь будет подробно разобран инструмент Trello. Он будет разобран со стороны общей информации и с более подробной стороны использования данных ресурсов на практике. Ниже будет представлен материал в том примерном виде, в котором его можно поместить на веб ресурс.

### Trello

Trello — облачная программа для управления проектами небольших групп, разработанная Fog Creek Software. Trello использует парадигму для управления проектами, известную как канбан.

Trello был представлен на мероприятии TechCrunch Disrupt основателем Fog Creek Джоэлом Спольски. В июле 2012 года количество посетителей сайта превысило 500 000 пользователей. К декабрю 2012 года превысило уже 1 000 000 пользователей. В январе 2017 года разработчик сервисов для совместной работы в IT-компаниях Atlassian приобрел Trello за \$425 млн.

Онлайн-сервис Trello – это универсальный инструмент для управления рабочими и личными проектами. Он позволяет отслеживать выполнение каждой задачи, координировать работу нескольких человек, следить за сроками и хранить всю необходимую информацию в одном месте. Пользователи ценят его за простоту и удобство, а также за практически неограниченный бесплатный доступ.

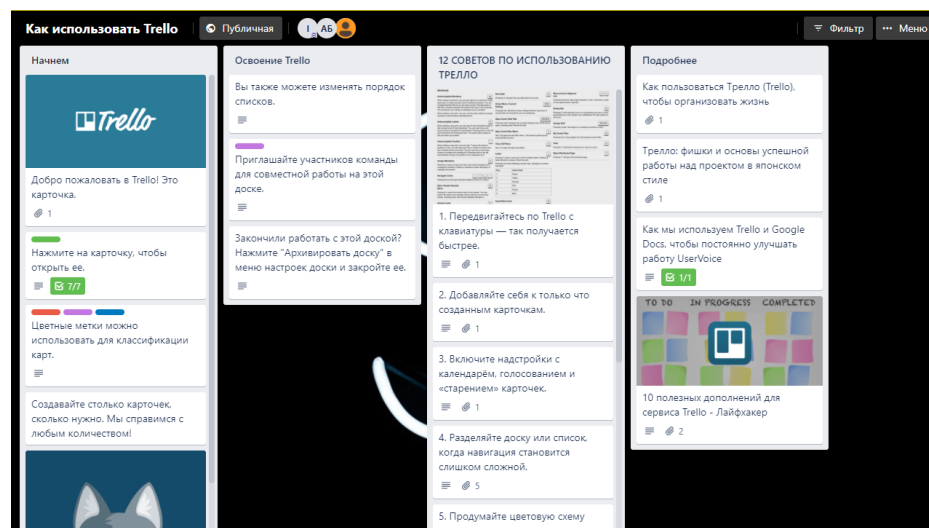


Рисунок 15. Примерная доска Trello

## 1) Планирование в Trello

Ключевой принцип работы «по канбану» — представить все задачи и их статусы максимально наглядно для всех участников рабочего процесса. В классическом виде это выглядит так: мастер цеха для каждой операции пишет карточку и вывешивает её на специальную доску. По мере выполнения операции карточка перемещается по доске, а затем снимается. Таким образом, каждый может узнать, кто и какую задачу выполняет — для этого достаточно просто взглянуть на доску.

Trello — это виртуальная доска для карточек и их перемещения. Сервис не ограничивает конкретным подходом к управлению задачами: можно применять любую гибкую методологию (например, Agile и Scrum) или придумать собственную систему.

## 2) Возможности Trello:

- Облачное решение. Всё хранится на облачных серверах, поэтому ничего не потеряется и не забудется.
- Все плюсы онлайн-сервиса. Отслеживание дедлайнов, уведомления, прикрепление файлов, поиск по задачам, фильтрация и сортировка, подробные логи и статистика.
- Кроссплатформенность. Сервис доступен в веб-версии (для Chrome, Firefox, Safari и Edge), есть приложение для компьютера, планшета и смартфона.
- Можно пользоваться бесплатно. Trello выгодно отличается от прочих продуктов почти неограниченными функциями в бесплатной версии. Единственное ограничение — нельзя загрузить файл тяжелее 10 Мб.

- Интеграция Trello с внешними сервисами. Trello можно связать с внешними сервисами, например, почтой, календарем, Github, Slack и другими. Trello можно подключить и к Tilda: тогда заявки, попадающие через формы Tilda, будут автоматически появляться на досках Trello.

### 3) Как пользоваться Trello.

Trello — это простая иерархия из досок, списков и карточек. В доске находятся списки, в списках — карточки.

#### - Доска задач Trello

Доска — это общее пространство, в котором можно размещать карточки, группировать их по колонкам и перемещать карточки между списками. В виде доски можно представить проект, работу одного отдела или целой небольшой компании, временной промежуток (например, неделя или год) или даже один процесс (например, написание диссертации).

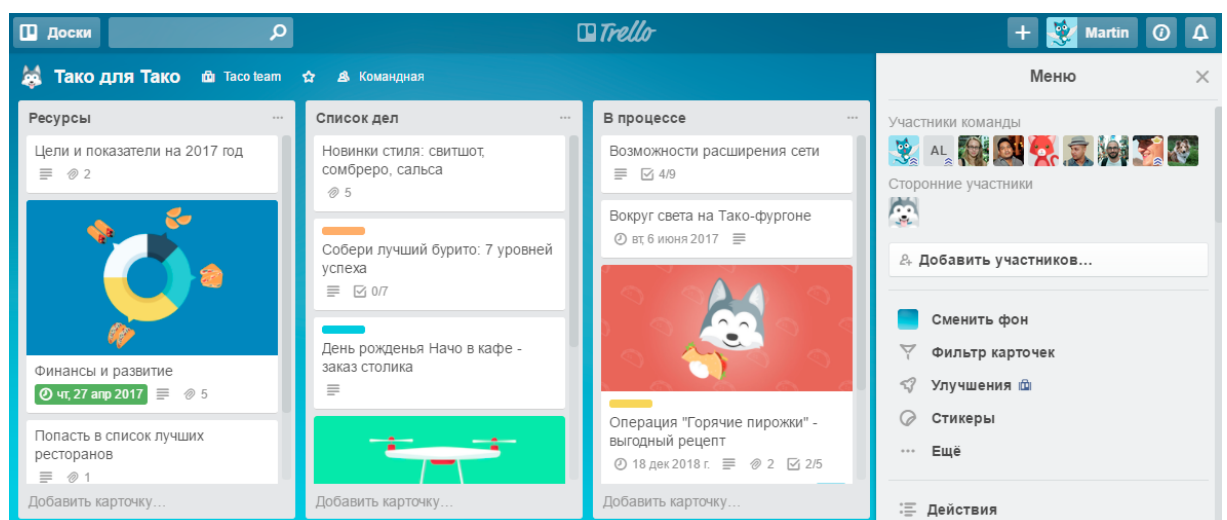


Рисунок 16. Доска Trello

#### - Что можно делать с доской?

1. Копировать. Если процессы одного проекта подходят и для другого, не нужно заводить доску заново. Достаточно просто скопировать существующую и немного её перенастроить.

2. Добавить к ней других участников: коллег, друзей, членов семьи, экспертов.

3. Добавить улучшения. «Улучшениями» в Trello называются плагины, которые расширяют функциональность: переключение на календарный вид или интеграция с Google картами. В бесплатной версии доступно одно любое улучшение.

4. Изменить фон доски. Его не очень хорошо видно, когда колонок и карточек много, но он помогает отличить один проект от другого на домашней странице Trello.

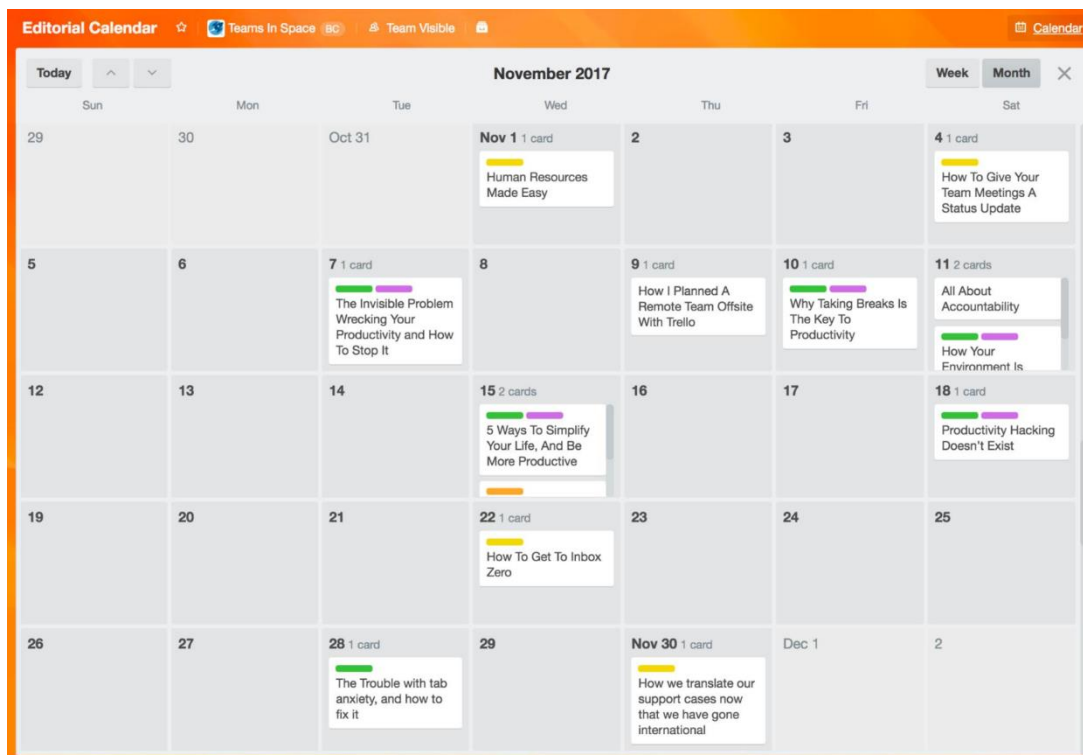


Рисунок 17. Календарь Trello

На сайте Trello есть шаблоны досок под разные задачи. Все шаблоны находятся на домашней странице в отдельной вкладке и разбиты по категориям: «Бизнес», «Дизайн», «Образование», «Личные дела» и так далее.

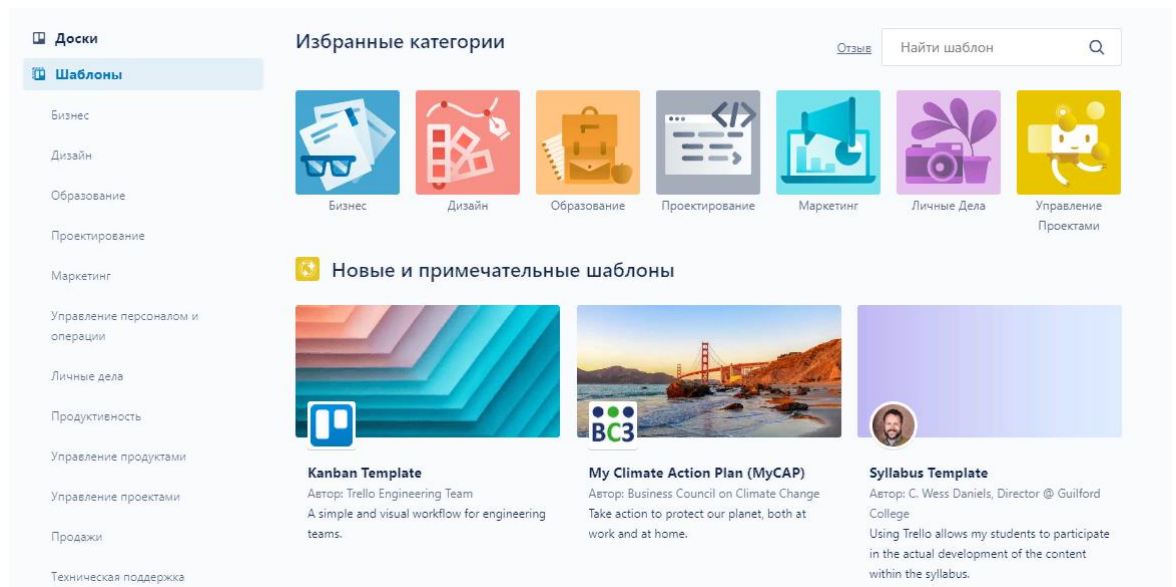


Рисунок 18. Шаблоны досок

Доски могут быть приватными, командными и публичными.

Приватная — доступна только вам и тем, кого вы пригласили.

Командная — доступна для команды, которую вы создали.

Публичная — доступна любому пользователю интернета. Она индексируется поисковыми системами — её могут найти по соответствующему запросу. Вот пример открытой доски.

- Списки (или колонки) Trello.

Доска не имеет смысла, если в ней нет списков-колонок. Именно по ним будут путешествовать задачи в ходе выполнения.

Обычно в качестве колонок выступают:

- Статусы задач. Например, «Сделать», «В работе», «В доработке», «Сделано».

- Время. Например, «Вчера», «Сегодня», «Завтра», «Когда-нибудь потом» или «Январь», «Февраль», «III квартал 2020».

- Приоритет. Например, «Срочно», «Выполнить в течение 2 дней», «Выполнить в течение недели».

- Виды задач. Например, «SEO», «Дизайн», «Контент», «Разработка».

Этапы. Например, «Первый звонок», «Второй звонок», «Встреча», «Сделка», «Лиды».

#### 4) Карточки Trello

Карточки — базовый элемент Trello. Именно в них кипит работа: появляются сроки и метки, добавляются участники, ведутся обсуждения. Обычно карточка — это одна задача большого проекта.

- Какая информация есть в карточке?

1. Название. Например, «Добавить всплывающее окно на главную».
2. Описание. Например, «Всплывающее окно с предложением подписаться на почтовую рассылку. Активируется при скролле (75%)».
3. Участник или исполнитель. Их может быть несколько человек.
4. Срок. Например, 23 февраля 21:00. При добавлении срока в карточке появляется чек-бокс о выполнении.
5. Чек-лист. Например, «1. Получить текст от копирайтера. 2. Создать попап. 3. Написать маркетологу».
6. Обложка. Если добавить к карточке изображение, оно автоматически станет обложкой и выделит карточку в колонке. Это можно отменить в настройках карточки.
7. Метка. Метки — это цветные ярлыки для задач. Ярлыками можно разделять задачи по их типу, сложности, важности, отделу или вашей личной симпатии к задаче.
8. Комментарии. Комментарий к карточке может оставить любой участник доски. Например, «Не успею выполнить к 21:00». Чтобы упомянуть участника, введите его имя через @.

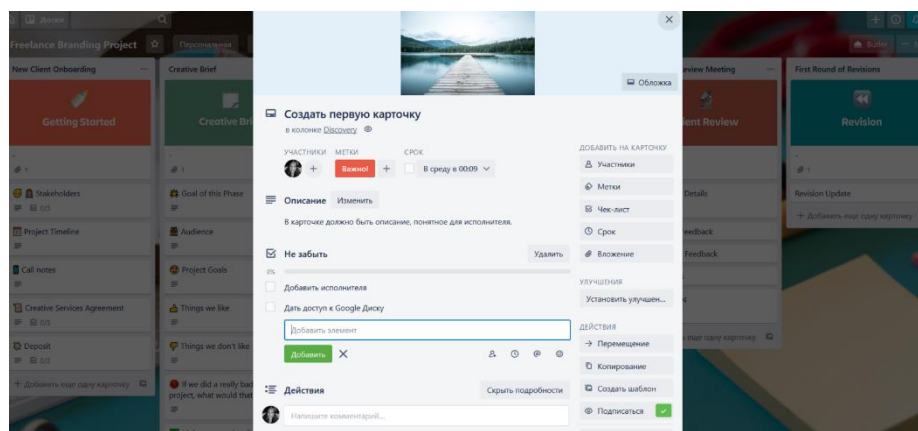


Рисунок 19. Пример создания карточки



## - Шаблоны Trello

При наличии большого количества однотипных карточек, можно создать шаблон. Для этого надо выбрать в меню карточки «Создать шаблон» — и на ней появится специальная метка.

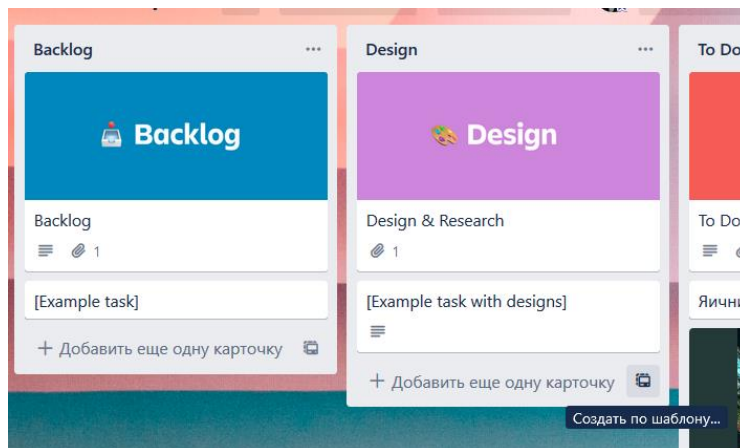


Рисунок 20. Пример создания карточки по шаблону

## - Дата выполнения

Если в карточке установлен срок выполнения, отметка с датой станет жёлтой и красной по мере приближения к дедлайну. Если задача выполнена в срок и отмечена как выполненная, отметка с датой будет зелёной. Эта небольшая деталь выступает хорошим психологическим крючком для своевременного исполнения задачи.

## - Метки Trello

Метки — это мощный инструмент, который в отдельных случаях может заменить собой списки. Например, метки со статусами задач покажут, на каком этапе сейчас находится процесс. Однако это достаточно редкая практика, которая подходит для линейной работы в небольшой команде.

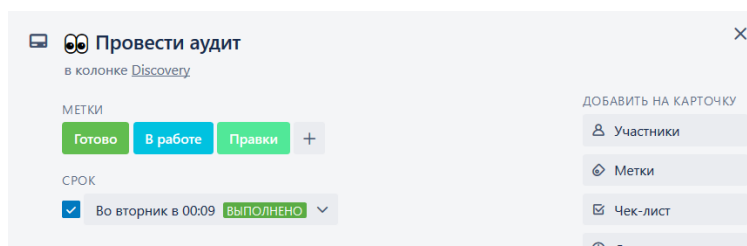


Рисунок 21. Пример меток



## - Чек-лист Trello

Чек-листы необходимы, если задача состоит из более мелких этапов. Например, в задаче по созданию лендинга необходимо написать текст, подобрать картинки и сверстать страницу. Каждая ступень будет отдельным пунктом чек-листа. Прогресс можно увидеть, даже не открывая карточку.

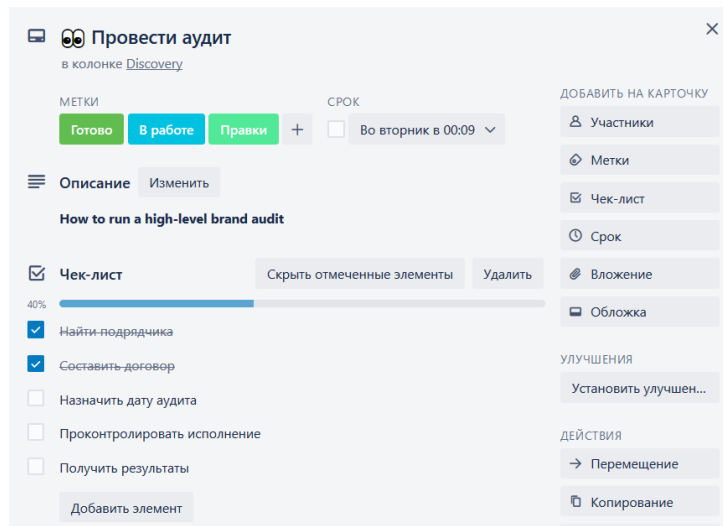


Рисунок 22. Пример чек-листа

## - Перемещение между колонками

Карточки легко перетащить из колонки в колонку: достаточно одного взгляда на доску, чтобы оценить, как идут дела на проекте или в отделе.

Перемещать карточку следует в тот момент, когда появляются изменения: например, выполнение задачи перешло на другой этап, она передана другому человеку, стала неактуальна или работа над ней завершена.

## 5) Примеры использования: чем можно управлять с помощью Trello

- Личный планировщик
- Список дел
- Планирование мероприятия
- Контент-план
- Баг-трекер
- Мониторинг проблем в бизнес-процессах

- Трекер задач для технической поддержки
- Воронка продаж
- Подбор персонала
- Адаптация новых сотрудников
- Стратегическое планирование

### 2.3 Способы создания веб-ресурса

В данной главе будут рассмотрены примерные способы создания веб-базируемого информационного ресурса. Этот процесс будет разобран для трёх возможных случаев: при использовании сервиса Google Sites, при использовании сервиса Blogger, или при самостоятельном написании кода.

В данном разделе имеет смысл рассматривать способы создания сайта от самого простого, до самого сложного, поэтому начнём с Google Sites.

#### Google Sites

Самый простой способ создания сайта, так как он обладает блоковой структурой и зачастую использует метод drag and drop. Для работы с данным сервисом не понадобится знание кода, редактировать разделы, добавлять новые части сайта и новые страницы можно самостоятельно, при помощи интерфейса.

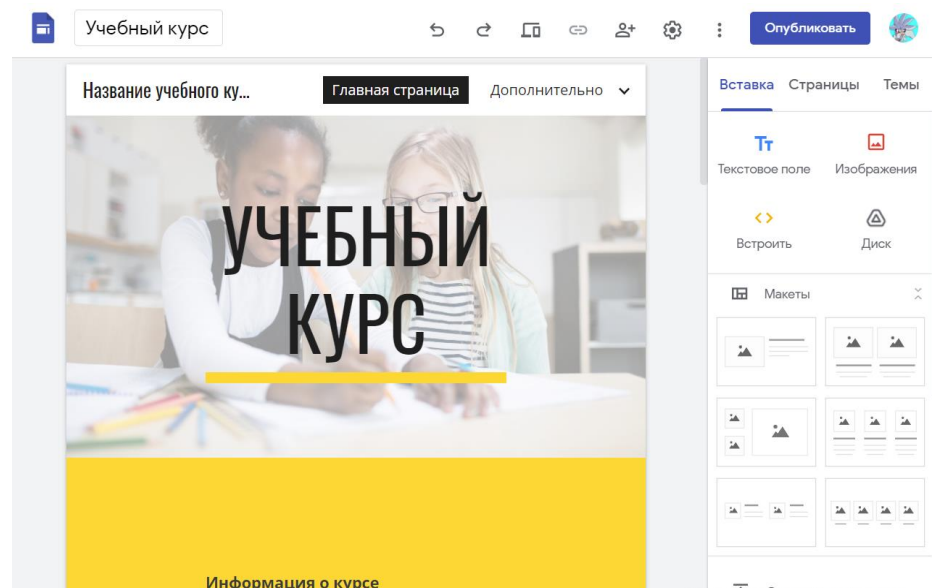


Рисунок 23. Конструктор Google Sites

Отформатируем Главную страницу, чтобы она подходила под надобности ресурса.

В итоге после внесённых изменений, добавления необходимой информации о управлении проектами, небольшого изменения дизайна, выйдет следующий результат.

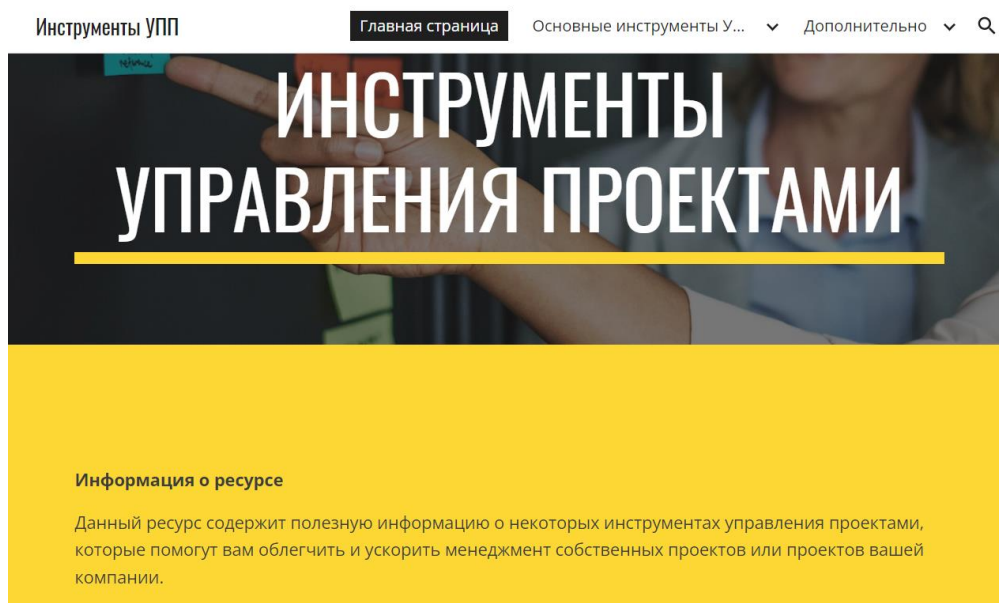


Рисунок 24. Главная страница на Google Sites (1)



Рисунок 25. Главная страница на Google Sites (2)

## ЧТО ТАКОЕ УПП?

Управление разработкой программного обеспечения — особый вид управления проектами, в рамках которого происходит планирование, отслеживание и контроль за проектами по разработке программного обеспечения. Ключевым моментом в управлении проектом по разработке программного обеспечения является правильный выбор метода разработки.

Основные отличия от других видов управления проектами

- Конечный результат проекта по разработке программного обеспечения нематериален
- Недостаточность накопленного в данной области опыта
- Быстрое изменение используемых в проекте технологий
- Опыт управления проектами по разработке программного обеспечения часто не может быть применён к другим проектам

Рисунок 26. Главная страница на Google Sites (3)

Также, для примера, проработаем примерную структуру сайта, которая была описана выше.

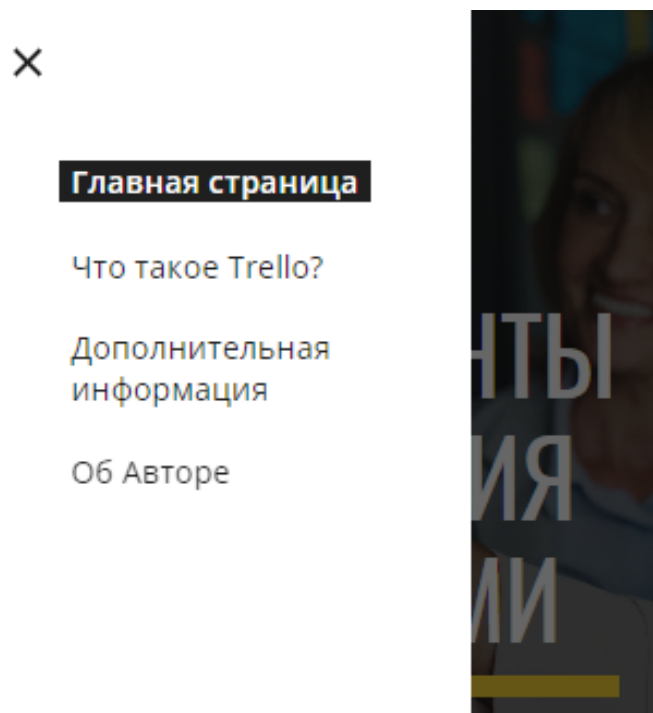


Рисунок 27. Панель навигации на Google Sites

Более подробная страница про инструмент Trello скрыта с панели поиска. В этот раздел можно попасть из раздела «Что такое Trello?». Так как этот раздел является одним из основных, его также следует проработать, по примеру «Главной страницы».

В итоге получим страницу, подобную приведённой ниже.



Рисунок 28. Страница «Что такое Trello?» на Google Sites

Здесь логотип ресурса, его название и текст с кратким описанием являются ссылками на страницу с информацией о данном ресурсе. Это единственный способ попасть на эту страницу. При этом, пользователь может перейти с этой страницы, на любую другую страницу, указанную в панели поиска.

Подобным образом, вставляя скриншоты, ссылки, текст, добавляя необходимые страницы и под страницы, новые разделы и блоки, можно проработать остальные страницы сайта.

По завершении процесса сайт можно опубликовать, используя внутренние возможности Google Sites.

## Публикация в Интернете

Веб-адрес

myamazingsitepleasedontsteal



<https://sites.google.com/view/myamazingsitepleasedontsteal>

### Пользовательский домен

Чтобы потенциальным посетителям было проще найти ваш сайт, создайте для него пользовательский домен, например [www.yourdomain.com](#).

[НАСТРОИТЬ](#)

У кого есть доступ к моему сайту

Все [НАСТРОИТЬ](#)

### Настройки поиска

☐ Запретить общедоступным поисковым системам показывать мой сайт. [Подробнее...](#)

[Отмена](#)

[Опубликовать](#)

Рисунок 29. Публикация через Google Sites

Таким образом, используя возможности Google Sites, можно быстро создать и опубликовать простой вебсайт, со всей найденной информацией.

Далее следует рассмотреть ресурс Blogger, так как на нём немного сложнее создавать сайты, ведь ресурс предназначен скорее для написания блогов. Но у него также есть некоторые плюсы и преимущества, из-за которых его можно предпочесть созданию сайта с нуля.

## Blogger

Второй по простоте из выбранных способов создания веб-ресурса. В данном случае сайт будет представлять собой блог, в котором страницы, посвященные конкретным инструментам, будут представлять собой сообщения в блоге. От этого слегка изменится структура построения ресурса, но это позволит при желании добавлять новые разделы о новых инструментах с легкостью. Blogger, так же как и Google Sites, обладает блоковой структурой и удобным интерфейсом, что позволяет довольно быстро освоиться в работе с ним, и позволяет без знания кода создать собственный блог.

Для начала, используя выбранный шаблон, и встроенный конструктор, преобразуем главную страницу под нужды ресурса.

В итоге получим страницу следующего вида.

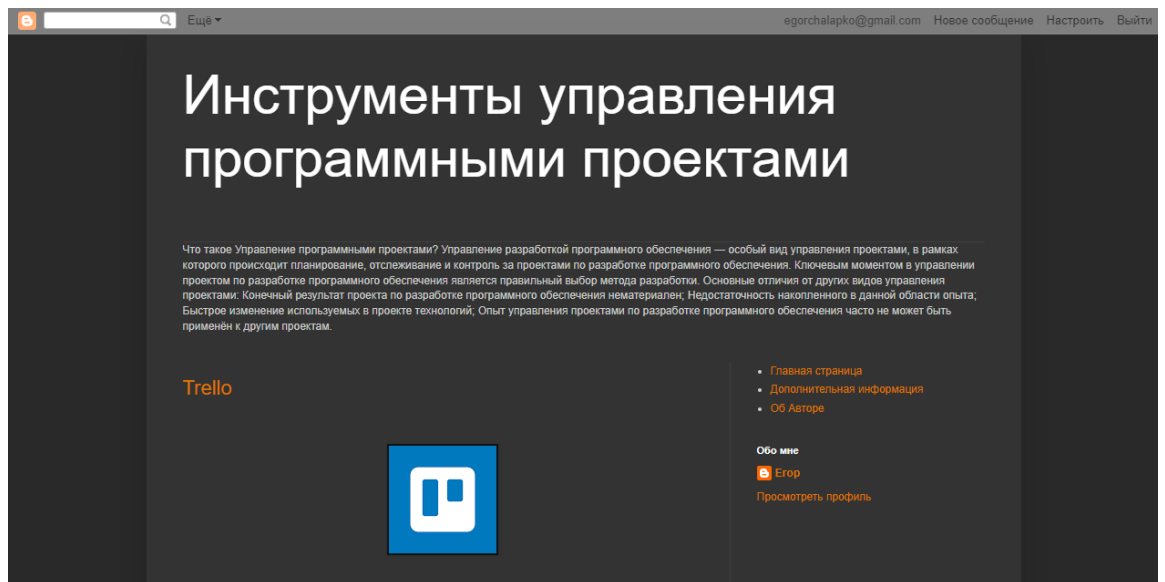


Рисунок 30. Главная страница на Blogger (1)

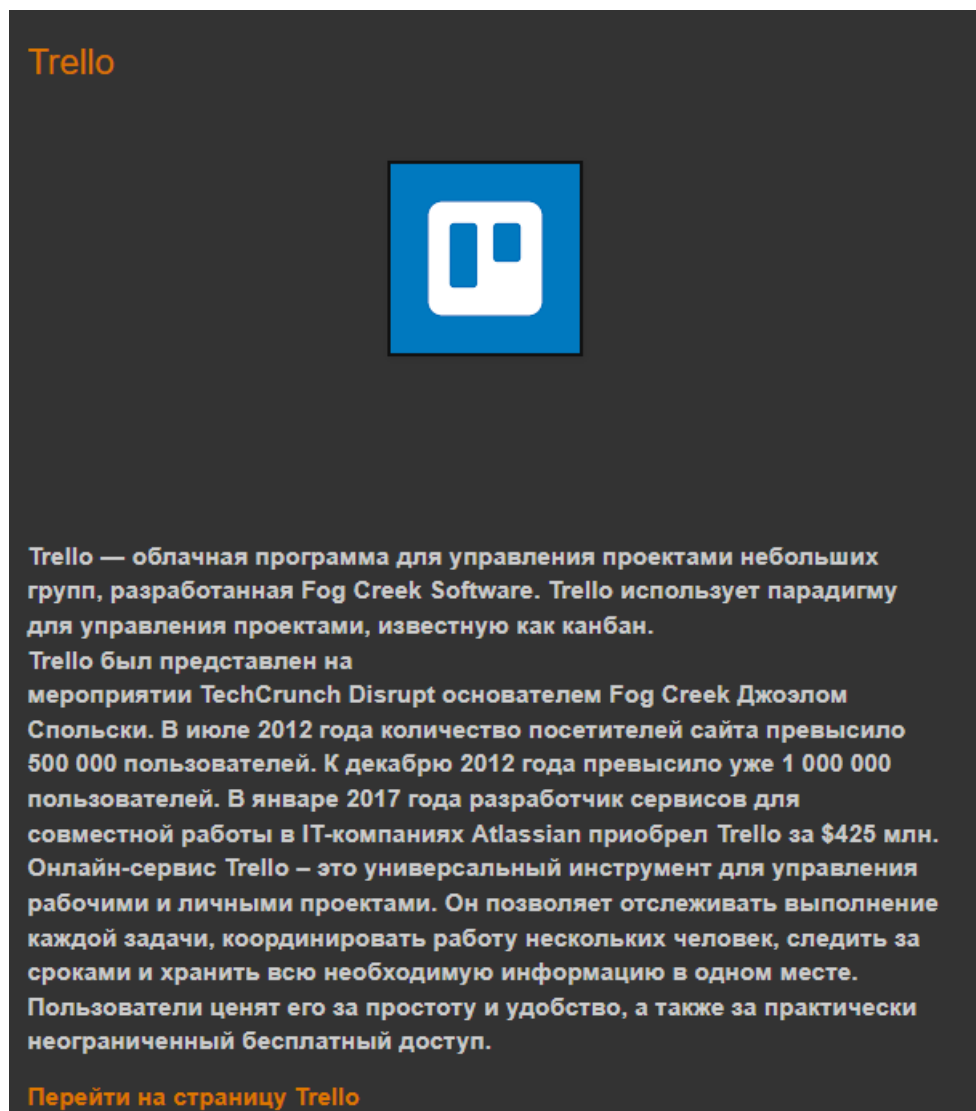


Рисунок 31. Главная страница на Blogger (2)

Данная страница имеет в себе краткое описание того, что такое УПП, а также сообщение о выбранном инструменте. В нём представлена краткая информация о ресурсе, а также возможность перехода на страницу с данным ресурсом, при этом страница скрыта с панели быстрого доступа. При этом с этой страницы можно перейти на любую доступную на панели.

Для ещё одного примера следует попробовать создать таким же образом страницу с дополнительной информацией.

В итоге получится страница подобная этой.

### Дополнительная информация

**Несколько интересных инструментов управления проектами**

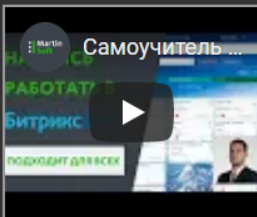
Asana — мобильное и веб-приложение для управления проектами в командах. Asana представляет собой SaaS-приложение для небольших компаний и индивидуального использования, обладающее всем необходимым набором возможностей для успешного ведения небольших проектов. Основной акцент создатели сервиса делают на том, что теперь управлять проектами возможно и без использования электронной почты. Концепция работы с приложением Asana проста и напоминает большинство систем подобного назначения. Каждая команда может создать для себя удобное рабочее пространство. Каждое рабочее пространство может включать в себя множество проектов, а каждый проект, в свою очередь, множество задач. Пользователи, имеющие на то полномочия, могут дополнять задачу, добавлять комментарии и теги, прикреплять к задаче файлы. Также Asana дает пользователям возможность подписаться на интересующие его задачи. В случае изменения или закрытия такой задачи все подписчики моментально получат соответствующее уведомление.

Slack — корпоративный мессенджер. Slack собирает в одном окне обсуждения в общих темах (каналах), приватных группах и личных сообщениях; имеет собственный хостинг, режим предпросмотра изображений и позволяет искать среди всех сообщений сразу. Кроме того, Slack поддерживает интеграцию с почти 100 сторонними сервисами, такими как Dropbox, Google Drive, GitHub, Google Docs, Google Hangouts, Twitter, Trello, MailChimp, Heroku, Jira. Slack доступен для пользователей следующих операционных систем:


- Windows
- macOS
- GNU/Linux
- Android
- iOS
- Windows Phone

**Полезные видеоуроки по теме**

Битрикс24



Самоучитель ...



CRM Битрикс ...

Рисунок 32. Страница «Дополнительная информация» на Blogger (1)



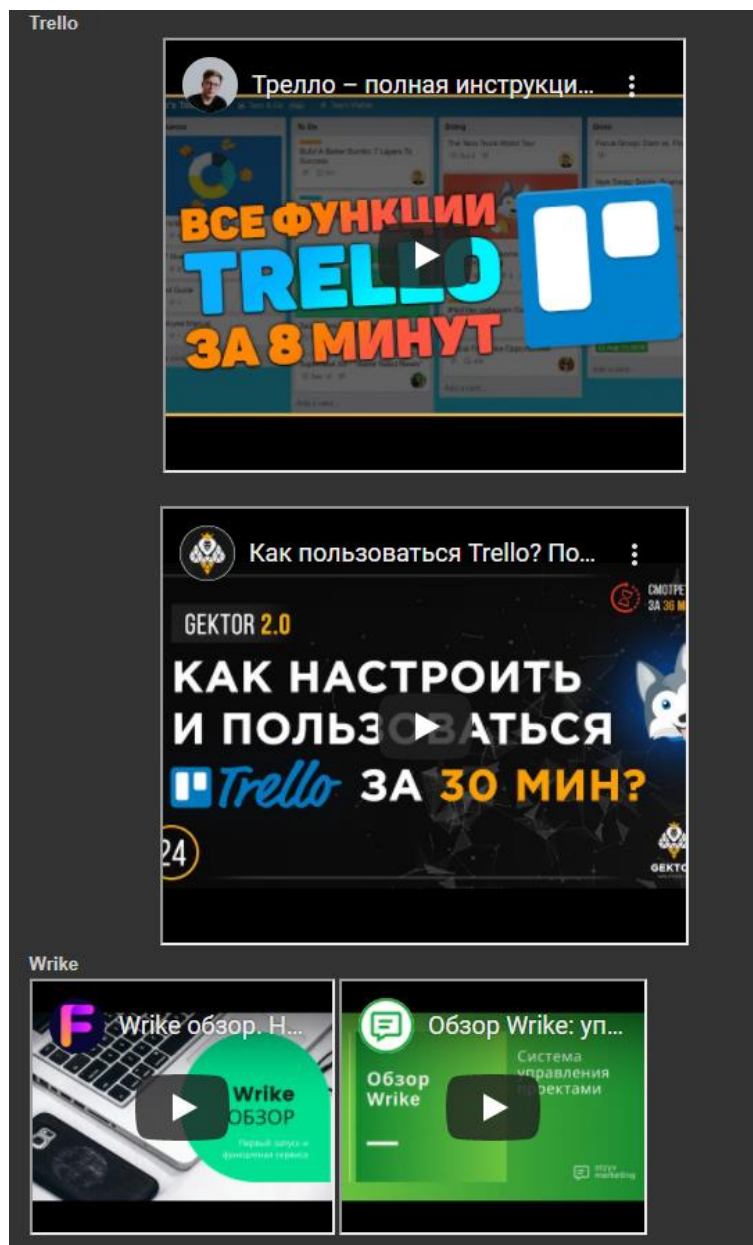


Рисунок 33. Страница «Дополнительная информация» на Blogger (2)

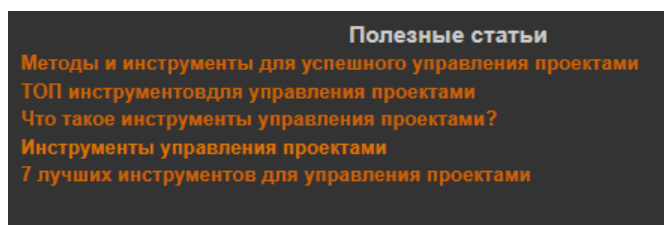


Рисунок 34. Страница «Дополнительная информация» на Blogger (3)

Таким же способом можно создать и остальные разделы сайта. Блог просто публикуется, благодаря встроенным возможностям Blogger.

Далее имеет смысл перейти к третьему и самому трудоёмкому способу создания сайта – написание собственного сайта на html. Данный процесс занимает наибольшее время, и

требует публикации созданных html файлов, а также знания кода.

## HTML

Данный способ, несмотря на вышеперечисленные минусы, имеет также и очень серьёзные плюсы. Зная разметку, можно создавать сайты любых форматов. Ограничивается всё только самим знанием кода и навыками создания сайта. Данный способ позволит создать максимально подходящий сайт с нуля или используя заготовленные шаблоны. В соответствии с предложенными шаблонами, следует написать код для главной страницы сайта.

В итоге получим нечто на подобии этой страницы.

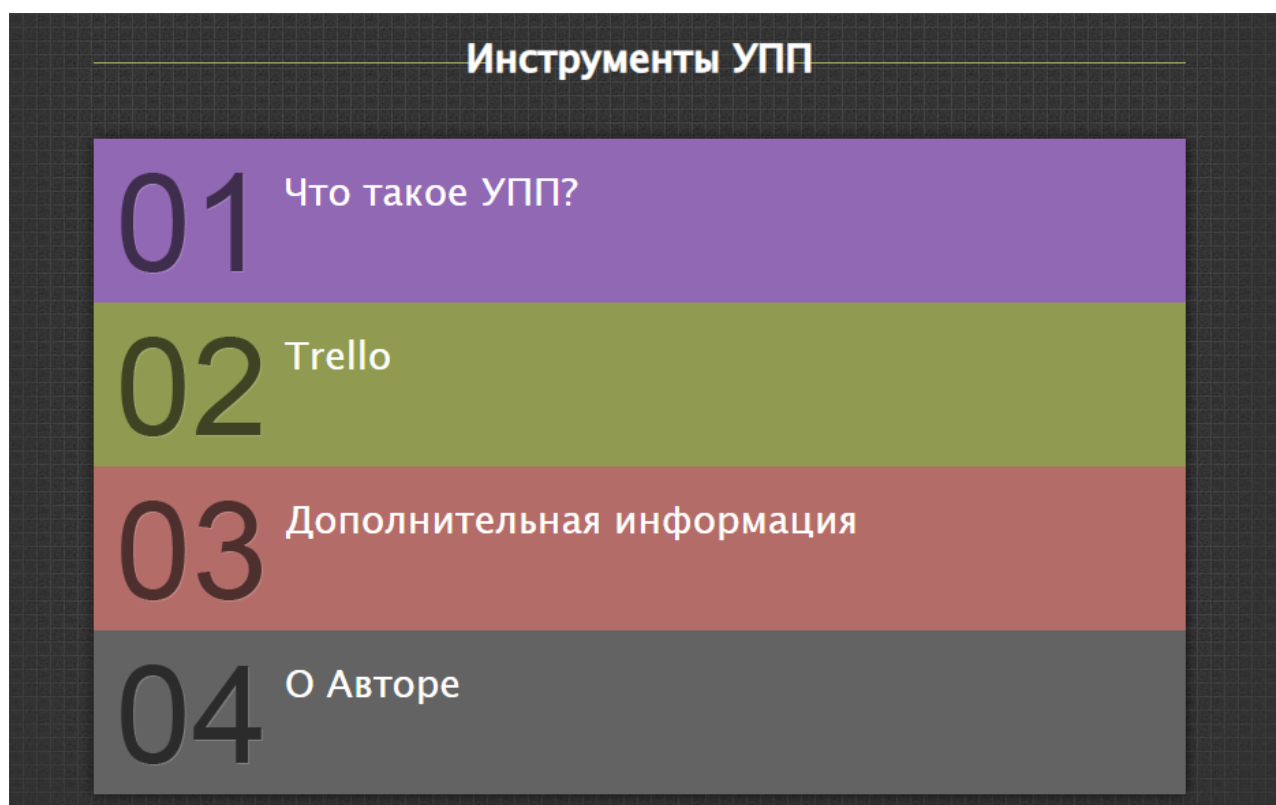


Рисунок 35. Главная страница на HTML

```

26 <div id="templatemo_header_wrapper">
27   <div id="templatemo_header">
28     <div id="site_title"><a href="#">Инструменты УПП</a></div>
29   </div>
30 </div>
31
32 <div id="templatemo_main_wrapper">
33   <div id="templatemo_main">
34     <div id="content">
35       <div id="home" class="section">
36         <div class="home_box left">
37           <div class="row1 box box1">
38             <div class="box_with_padding">
39               <h2><a href="#about">Что такое УПП?</a></h2>
40             </div>
41           </div>
42           <div class="row1 box2">
43             <div class="box_with_padding">
44               <h2><a href="#services">Trello</a></h2>
45             </div>
46           </div>
47           <div class="row1 box3">
48             <div class="box_with_padding">
49               <h2><a href="#addinf">Дополнительная информация</a></h2>
50             </div>
51           </div>
52           <div class="row1 box4">
53             <div class="box_with_padding">
54               <h2><a href="#contact">О Авторе</a></h2>
55             </div>
56           </div>
57         </div>
58       </div>
59     </div>
60   </div>
61 </div>

```

Рисунок 36. Код главной страницы на HTML

Данная страница содержит переходы на остальные страницы с более подробной информацией. Следует рассмотреть, как осуществляется переход на другую страницу, как она выглядит, и как она устроена. Рассмотрим на примере страницы «Что такое УПП»

```

60
61 <div class="section_section_with_padding" id="about">
62   <h1>Что такое УПП?</h1>
63   <div>
64     <p>Управление разработкой программного обеспечения – особый вид управления проектами, в рамках которого происходит планирование,
65     <p>Основные отличия от других видов управления проектами</p>
66     <ul>
67       <li>- Конечный результат проекта по разработке программного обеспечения нематериален</li>
68       <li>- Недостаточность накопленного в данной области опыта</li>
69       <li>- Быстрое изменение используемых в проекте технологий</li>
70       <li>- Опыт управления проектами по разработке программного обеспечения часто не может быть применён к другим проектам</li>
71     </ul>
72   </div>
73   <a href="#home" class="home_btn">home</a>
74   <a href="#home" class="page_nav_btn previous">Previous</a>
75   <a href="#services" class="page_nav_btn next">Next</a>
76 </div>
77

```

Рисунок 37. Код страницы «Что такое УПП?» на HTML

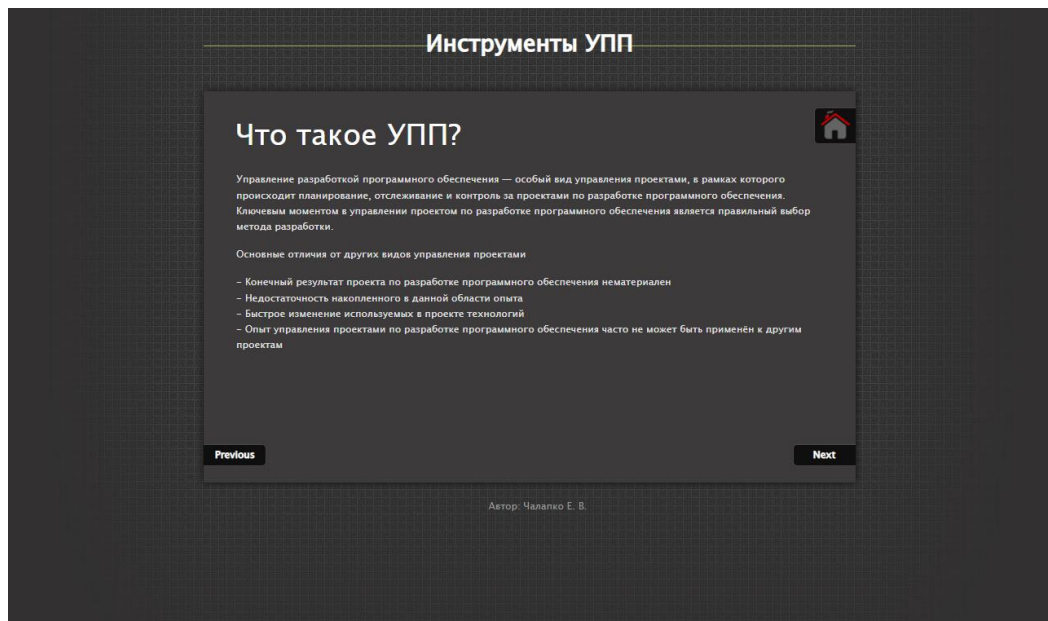


Рисунок 38. Страница «Что такое УПП?» на HTML

На странице располагается размещённая в соответствии с разметкой информация, а также 3 кнопки для навигации по ресурсу. Они перемещают пользователя в соответствии с номерами разделов на следующий («Next»), предыдущий («Previous») и на главную страницу (иконка дома), предавая ресурсу вид презентации.

Такой вид перемещения подходит для всех типов страниц, не считая страницы с подробным разбором ресурса, для данных страниц следует сделать отдельный доступ, но для них можно сохранить структуру презентации.

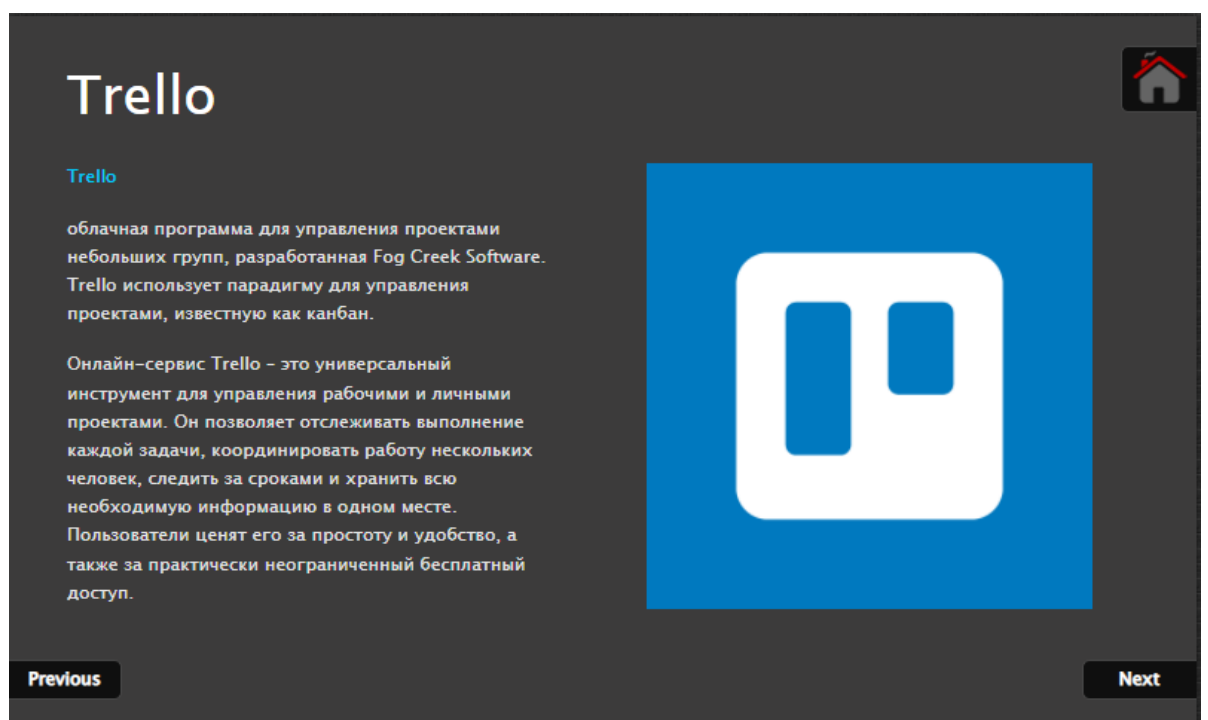


Рисунок 39. Страница «Trello» на HTML

```

79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94

<div class="section section_with_padding" id="services">
<h1>Trello</h1>
<div class="half left">
<p><a href="#trello">Trello</a></p>
<p>Облачная программа для управления проектами небольших групп, разработанная Fog Creek Software. Trello использует парадигму для управления проектами.
<p>Онлайн-сервис Trello – это универсальный инструмент для управления рабочими и личными проектами. Он позволяет отслеживать выполнение каждой задачи.
</div>
<div class="half right">
<p><a href="#trello"></a></p>
</div>
<a href="#home" class="home_btn">home</a>
<a href="#about" class="page_nav_btn previous">Previous</a>
<a href="#addinf" class="page_nav_btn next">Next</a>
</div>

```

Рисунок 40. Код страницы «Trello» на HTML

В данном разделе представлена краткая информация о ресурсе, возможность перейти на следующую страницу, вернуться на предыдущую или на главную страницы, но также, здесь можно перейти на начальную страницу, посвящённую конкретному инструменту УПП. Сделать это можно, перейдя по ссылке через название ресурса, или через его логотип.

Устройство самих страниц с подробным описанием будет примерно выглядеть вот таким образом.

```

136
137
138
139
140
141
142
143
144
145
146
147
148
149

<div class="section section_with_padding" id="trello">
<h1>Планирование в Trello</h1>
<div>
<p>Ключевой принцип работы «по канбану» – представить все задачи и их статусы максимально наглядно для всех участников рабочего процесса. В классическом
</div>
<div>
Trello – это виртуальная доска для карточек и их перемещения. Сервис не ограничивает конкретным подходом к управлению задачами: можно применять любую ги
</div>
<a href="#home" class="home_btn">home</a>
<a href="#services" class="page_nav_btn previous">Previous</a>
<a href="#trello2" class="page_nav_btn next">Next</a>
</div>

```

Рисунок 41. Код первой страницы про Trello на HTML



Рисунок 42. Первая страница про Trello на HTML

```

150 <div class="section section_with_padding" id="trello2">
151 <div>
152     Возможности Trello
153     <ul>
154         <li>- Облачное решение. Всё хранится на облачных серверах, поэтому ничего не потеряется и не забудется.</li>
155         <li>- Все плюсы онлайн-сервиса. Отслеживание дедлайнов, уведомления, прикрепление файлов, поиск по задачам, фильтрация и сортировка, подробные логи
156         <li>- Кроссплатформенность. Сервис доступен в веб-версии (для Chrome, Firefox, Safari и Edge), есть приложение для компьютера, планшета и смартфона.
157         <li>- Можно пользоваться бесплатно. Trello выгодно отличается от прочих продуктов почти неограниченными функциями в бесплатной версии. Единственное
158         <li>- Интеграция Trello с внешними сервисами</li>
159     </ul>
160 </div>
161 <a href="#home" class="home_btn">home</a>
162 <a href="#trello" class="page_nav_btn previous">Previous</a>
163 <a href="#trello3" class="page_nav_btn next">Next</a>
164 </div>
165
166
167
168
169
170

```

Рисунок 43. Код второй страницы про Trello на HTML

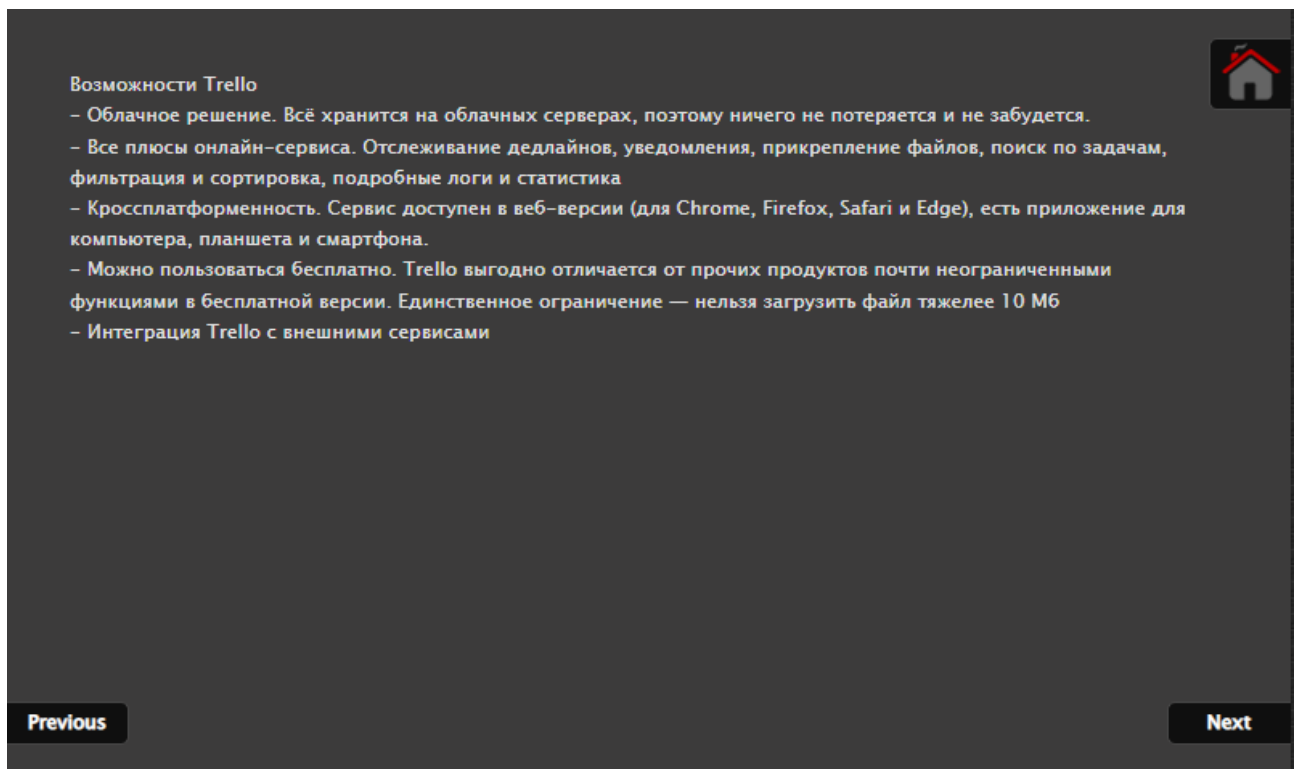


Рисунок 44. Вторая страница про Trello на HTML

Устройство данных страниц похоже на предыдущие, описанные выше, но теперь уже больше напоминают структуру презентации, так как осуществляется такой «переход между слайдами» с информацией. Подобным образом будут устроены и остальные подразделы для инструмента Trello.

Таким образом, в виде локальных файлов html и прочих, будет получен сайт, но для его использования потребуется его деплой, чтобы его можно было найти в интернете. Существует большое количество разных способов разместить сайт в сети, но одним из самых простых и доступных, является деплой в сервисе GitHub. GitHub позволяет хранить все

необходимые файлы в онлайн репозитории, и через сервис Github Pages разместить их в интернете на личном сайте, в том числе и html файлы, что позволит просматривать их, а также редактировать их в любое время. Сервис достаточно прост в освоении и удобен в использовании.

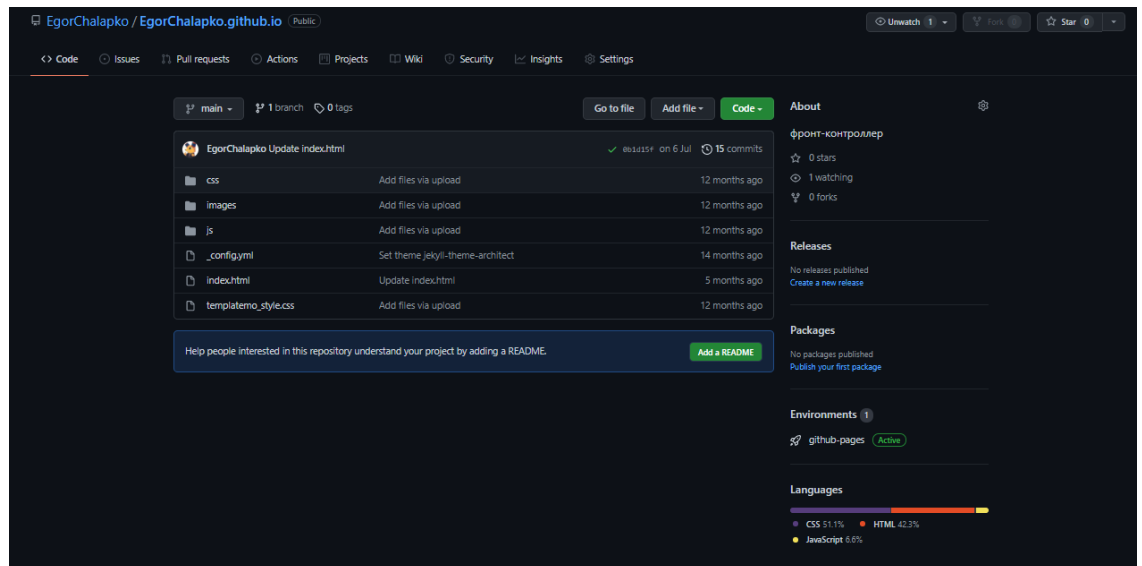


Рисунок 45. Репозиторий на GitHub

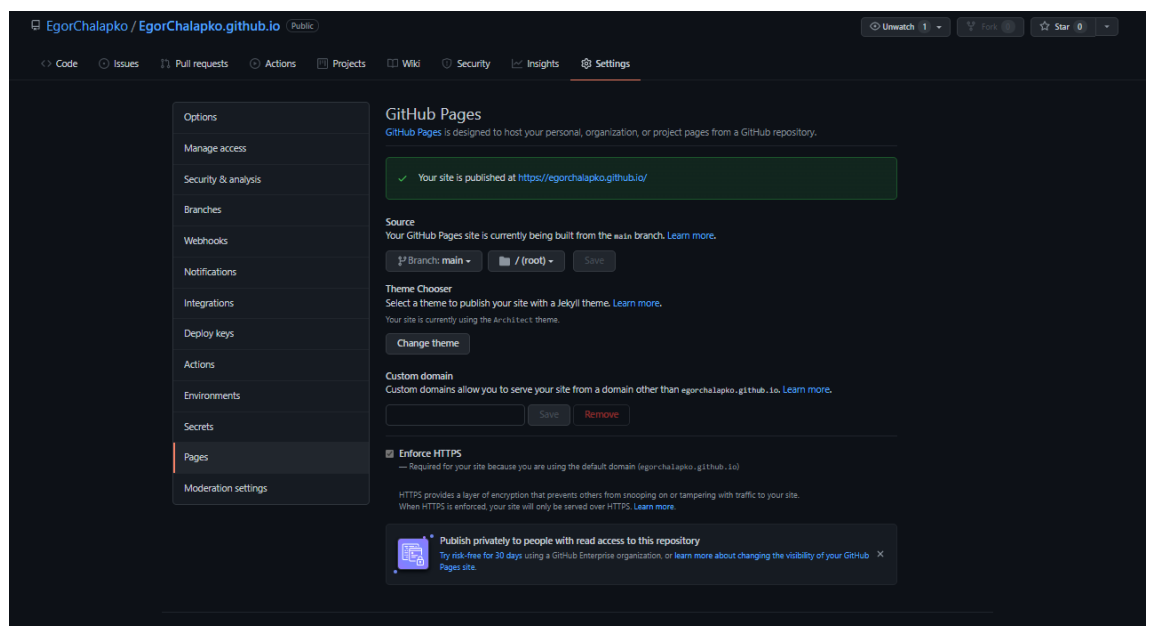


Рисунок 46. Раздел GitHub Pages

В итоге были рассмотрены с разных сторон все выбранные способы создания информационного ресурса.

## ЗАКЛЮЧЕНИЕ

В итоге данной курсовой работы были выполнены все поставленные задачи. Был подробно составлен план работы над проектом, возможные варианты его развития, дизайна и представления. Были проведены сбор, исследование и форматирование материалов, необходимых для выполнения поставленной задачи. Всё это позволило рассмотреть различные способы выполнения подобных задач, а именно – разработки информационного ресурса.

Данные ресурсы могут создаваться очень простыми способами для донесения информации до пользователя, при помощи ресурсов автоматической сборки сайтов, или же требовать огромное количество времени и сил, умения написания кода, дабы сделать этот сайт по-своему уникальным. Веб ресурс может принимать различные виды, будь то обычный вебсайт, постоянно обновляющийся сайт, построенный по модели блога, или же онлайн-презентация, по интересующей теме.

Помимо этого, также были исследованы инструменты УПП. Подобные ресурсы тоже предстают в разных формах, предоставляя различный функционал, будь то доски для менеджмента больших или собственных проектов, средства связи с работниками, отслеживание результатов работы проектов, и даже работы с клиентами или управления деятельностью самих компаний.

Во время работы над ресурсом были исследованы различные методологии разработки программного обеспечения, которые позволяют чётко работать над проектом, отслеживать все этапы разработки и чётко формулировать действия внутри рабочей команды. Среди рассмотренных методологий была выбрана одна, подходящая по большинству параметров данной разработке.

Результатом данной курсовой работы стал готовый рассмотренный проект, разработанный по выбранной указанной методологии разработки ПО.



## СПИСОК ЛИТЕРАТУРЫ

- 1) Методология Kanban: введение / Хабр [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/230725/>
- 2) Что такое Канбан-метод – максимально коротко — статья в блоге ScrumTrek [Электронный ресурс]. – Режим доступа: <https://scrumtrek.ru/blog/kanban/1360/chtotakoe-kanban-metod-maksimalno-korotko/>
- 3) Kanban: краткое знакомство | Atlassian [Электронный ресурс]. – Режим доступа: <https://www.atlassian.com/ru/agile/kanban>
- 4) Канбан (разработка) — Википедия [Электронный ресурс]. – Режим доступа: [https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D0%BD%D0%B1%D0%B0%D0%BD\\_\(%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0\)](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D0%BD%D0%B1%D0%B0%D0%BD_(%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0))
- 5) Модели и методологии разработки ПО | GeekBrains - образовательный портал [Электронный ресурс]. – Режим доступа: <https://gb.ru/posts/methodologies>
- 6) Ещё раз про семь основных методологий разработки / Хабр [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/edison/blog/269789/>
- 7) Выбор методологии разработки ПО: ищем верный подход [Электронный ресурс]. – Режим доступа: <https://issoft.by/blog/podkhody-k-razrabotke-po-kak-pravilno/#:~:text=%D0%9C%D0%B5%D1%82%D0%BE%D0%B4%D0%BE%D0%B%D0%BE%D0%B3%D0%B8%D1%8F%20%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%9F%D0%9E%20%E2%80%93%D1%8D%D1%82%D0%BE%20%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0,%D1%80%D1%83%D0%BA%D0%BE%D0%B2%D0%BE%D0%B4%D0%B8%D1%82%D0%B5%D0%BB%D1%8F%20%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B0%20%D0%B8%20%D0%B5%D0%B3%D0%BE%20%D0%BA%D0%BE%D0%BC%D0%B0%D0%BD%D0%B4%D1%8B.>
- 8) Методы и инструменты для успешного управления проектами [Электронный ресурс]. – Режим доступа: [https://4dru.com/blog/default/view?slug=methodu\\_i\\_instrumentu\\_dlya-uspeshnogo-upravleniya-proektami](https://4dru.com/blog/default/view?slug=methodu_i_instrumentu_dlya-uspeshnogo-upravleniya-proektami)
- 9) 20 лучших инструментов для менеджмента проектов - Kislod.io [Электронный ресурс]. – Режим доступа: <https://kislod.io/prosto-o-slozhnom/20-luchshih-instrumentov-dla-menedzhmenta-projektov/>

- 10) 10 инструментов для управления проектами: эффективность, постановка задач и общение — статьи на Skillbox / Skillbox Media [Электронный ресурс]. – Режим доступа: [https://skillbox.ru/media/management/10\\_instrumentov\\_dlya\\_PM/](https://skillbox.ru/media/management/10_instrumentov_dlya_PM/)
- 11) Как работать в Trello [Электронный ресурс]. – Режим доступа: <https://tilda.education/tpost/myda58x61h-trello-upravlenie-zadachami>
- 12) Что такое Trello и как им пользоваться | Медиа Нетологии [Электронный ресурс]. – Режим доступа: <https://netology.ru/blog/trello>
- 13) Trello [Электронный ресурс]. – Режим доступа: <https://trello.com/ru>
- 14) 7 лучших инструментов для управления проектами - Affde Marketing [Электронный ресурс]. – Режим доступа: <https://www.affde.com/ru/7-best-project-management-tools.html>
- 15) Что такое инструменты управления проектами? [Электронный ресурс]. – Режим доступа: <https://www.wrike.com/ru/project-management-guide/faq/instrumenty-upravleniya-proektami/>
- 16) ТОП инструментов для управления проектами - Блог системы управления проектами Worksection [Электронный ресурс]. – Режим доступа: <https://worksection.com/blog/top-management-tools.html>