

СИСТЕМЫ СБОРКИ\*, ПРОГРАММИРОВАНИЕ\*, OPEN SOURCE\*, JAVA\*, GITHUB

## GitHub Pages для сайта на 8 марта

igor\_suhorukov 8 марта 2017 в 12:17 👁 24,1k

Что такое HTML, CSS, JavaScript и Bootstrap фреймворк сложно объяснять человеку который далек от IT. А что если нужен сайт на бесплатном хостинге, без привязки к онлайн конструктору именно этого хостинга?!



Как думаете, реально научить девушку обновлять свой сайт на GitHub Pages к Международному женскому дню?! Расскажу как

мне это удалось с помощью генератора сайта, который доступен на [Github](#) и написан на Java + FreeMarker, к тому же старался автоматизировать публикацию контента в git репозиторий.

На днях помог осуществить мечту дизайнера по текстилю в интерьере о своем сайте, пока у меня уйма свободного времени. Она и сама пробовала создать макет с помощью конструктора на Wix. Но первый викс сайт комом вышел и нормально не отображался на мобильных телефонах... К тому же лучше не привязываться к конкретному хостингу.

Последнее что я верстал самостоятельно в HTML — был сайт художника во время учебы в университете и тогда была в моде табличная верстка. Веб разработка — не мой хлеб. А сейчас повсюду мобильные браузеры и блочная верстка. Барышня просто хотела сделать простой сайт-визитку, поэтому скрипты на сервере не нужны. При этом идеально было бы, чтобы обновляла фотографии и редактировала разделы этого веб ресурса с нулевым знанием HTML и без моей помощи.

Если ставить CMS — систему управления содержимым, то хостинг сайта будет дороже и придется тратить время на настройку, шаблоны и безопасность. Слишком сложно для статичного сайта. Мне показалось все это очень похоже на задачу генерации сайта по шаблону. Не знал где найти в наши дни хостинг для сайта без рекламы. Поспрашивав у знакомых, посмотрев варианты и почитав заметки в сети, пришел к выводу что GitHub Pages — идеальный вариант. Бесплатный, версионизируемый и поддерживает использование своего доменного имени для сайта.

Для GitHub Pages идеально подходит шаблонизатор [jekyll](#), но разбираться с ним и настройкой на компьютере дизайнера Ruby и jekyll gem совсем не было времени. С jekyll я поиграюсь в другой раз, как будет нужен сайт для своего проекта на GitHub.

Для любого джависта ассоциация с шаблонами — FreeMarker, знакомый многим шаблонный движок. Мне нужно было быстро разработать сайт и генератор, учитывая что Java мой привычный инструмент в работе. Сравнивая FreeMarker с Velocity, первый до сих пор жив и выглядит более функциональным.

Начали с создания [аккаунта для дизайнера](#) на GitHub:

The screenshot shows the GitHub profile page for user [nadinbox89](#). The browser address bar shows the URL <https://github.com/nadinbox89>. The profile header includes the GitHub logo, navigation links (Features, Explore, Pricing), a search bar, and links to sign in or sign up. The profile picture is a portrait of a woman with blonde hair. The username [nadinbox89](#) is displayed below the picture, along with the text "Joined 9 days ago". The profile statistics show 2 repositories, 6 stars, 0 followers, and 0 following. The "Popular repositories" section lists two repositories: [nadinbox89.github.io](#) (HTML) and [site](#) (HTML). The "23 contributions in the last year" section shows a calendar grid with green squares indicating contributions. The "Contribution activity" section shows a bar chart for March 2017, with a legend indicating 17 commits for [nadinbox89/nadinbox89.github.io](#) and 3 commits for [nadinbox89/site](#).

Потом перешли к дизайну и верстке. [Bootstrap Carousel](#) показался

отличным решением для показа изображений, так как основным содержимым веб страниц будут фото из портфолио и просматривать страницы будут на мобильных устройствах.

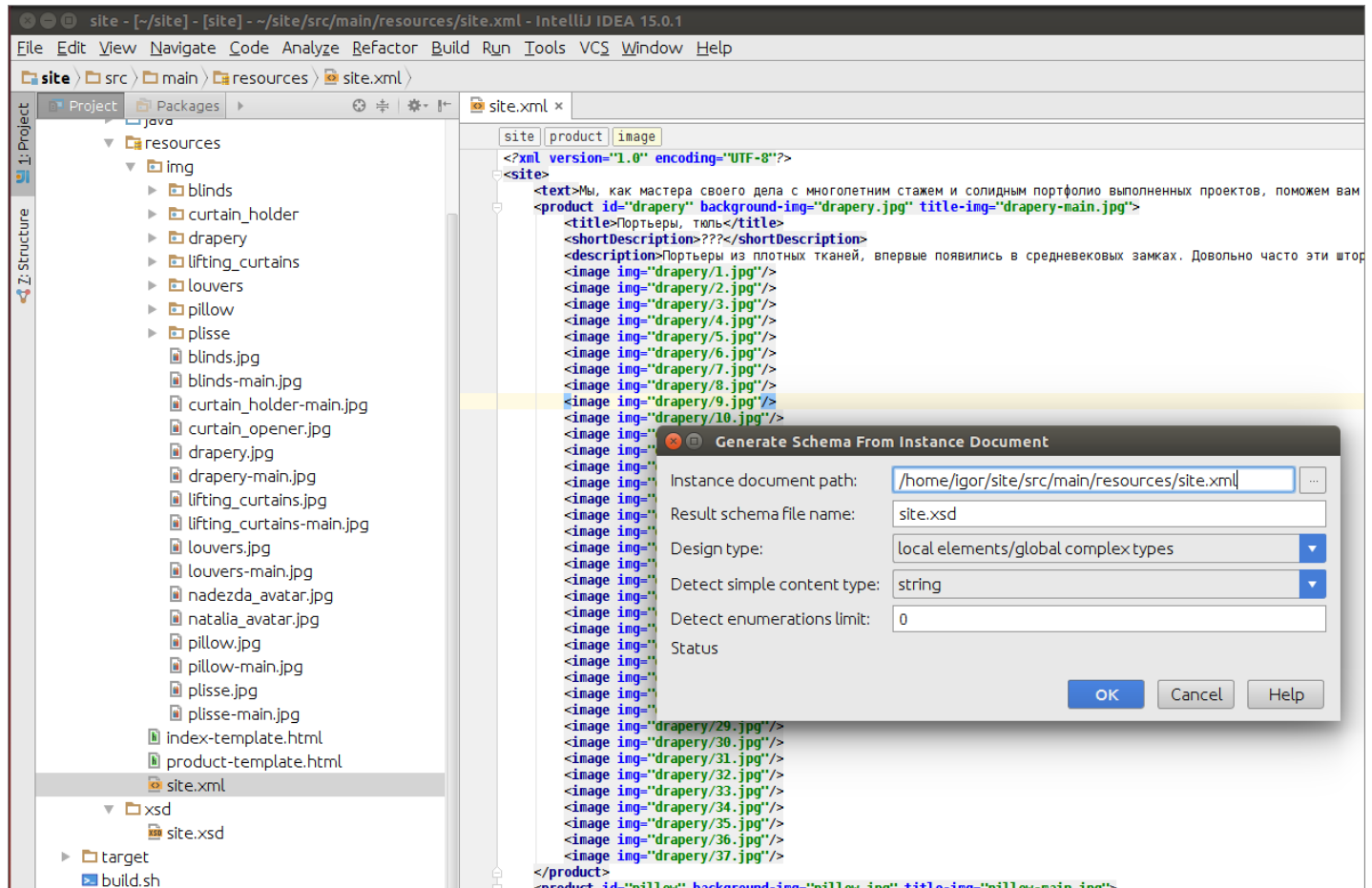
Раз уж подключил к странице Bootstrap, пригодятся элемент Panel и иконки для навигации/контактов. Для меня верстка была самым нудным этапом в создании сайта. На втором месте по сложности было [изменение размера, цветокоррекция и кадрирование большого числа фото](#) для портфолио работ.

Следующий шаг — превращение верстки в шаблон генератора. В шаблоне [FreeMarker](#) итерации по коллекции объектов делаются с помощью директивы `<#list product as prodItem>...</#list>`, а условия с помощью `<#if prodItem?is_first>...</#if>`. Это все что используется в этих шаблонах генератора.

За основу формата данных для хранения данных выбрал XML разметку. Со структурой быстро определились, на основе макета:

```
<site>
  <text>.....</text>
  <product id="drapery" background-img="drapery.jpg" title-
img="drapery-main.jpg">
    <title>Портьеры, тюль</title>
    <description>Портьеры из плотных тканей...</description>
    <image img="drapery/1.jpg"/>
    ...
    <image img="drapery/37.jpg"/>
  </product>
  ...
  <product .../>
</site>
```

Из XML файла сгенерировал **XSD schema** в IntelliJ Idea Community Edition.



Классы с аннотациями JAXB для использования в шаблоне генерируются динамически из Maven с помощью **jaxb2-maven-plugin**. При сборке проекта эти классы размещаются в директории **target/generated-sources/jaxb**.

Пример фрагмента **FreeMarker шаблона** для вывода всех изображений портфолио для определенной категории:

```
<#list image as imageInfo>
<div class="item<#if imageInfo?is_first> active</#if>">
  
```



```

        <div class="carousel-caption">
        <#if imageInfo.title??>
            <h3>${imageInfo.title}</h3>
            <p></p>
        </#if>
        </div>
    </div>
</#list>

```

А вот и пример данных, на основе которых генерируется страница:

```

    <product id="pillow" background-img="pillow.jpg" title-
img="pillow-main.jpg">
        <title>Покрывала, подушки</title>
        <shortDescription>Неотъемлемая часть текстильного
декорирования Вашей спальни, придающее уют, изысканность и особенный
стиль пространству</shortDescription>
        <description>Роскошное покрывало - это неотъемлемая часть
текстильного декорирования Вашей спальни, придающее уют,
изысканность и особенный стиль пространству. Со вкусом подобранные
ткани и фурнитура, которые сочетаются в шторах, покрывале и
подушках, создают гармонию и целостность Вашему интерьеру.
</description>
        <image img="pillow/1.jpg"/>
        <image img="pillow/2.jpg"/>
        <image img="pillow/3.jpg"/>
        <image img="pillow/4.jpg"/>
        <image img="pillow/5.jpg"/>
        <image img="pillow/6.jpg"/>
        <image img="pillow/7.jpg"/>
        <image img="pillow/8.jpg"/>
        <image img="pillow/9.jpg"/>
        <image img="pillow/10.jpg"/>
        <image img="pillow/11.jpg"/>
        <image img="pillow/12.jpg"/>
        <image img="pillow/13.jpg"/>
        <image img="pillow/14.jpg"/>
    </product>

```

В index.html генерируется оглавление на основе всех перечисленных в [site.xml](#) тегов <product...> с помощью [шаблона](#).

Как «клей» для шаблонов, основную работу выполняет java класс [src/main/java/Site.java](#) и maven [скрипт сборки](#): происходит копирование изображений из директории `src/main/resources/img` в `target/site`, из XML схемы генерируются классы с JAXB аннотациями и запускается программа на java с помощью `exec-maven-plugin`, которая с помощью FreeMarker и данных из [site.xml](#) генерирует сайт. Для плагина `jaxb2-maven-plugin` важно, чтобы `хjc` из JDK был доступен в одном из путей в переменной окружения `PATH`.

```
public class Site {
    public static void main(String[] args) throws Exception {
        File targetDirectory = getTargetDirectory();
        SiteType siteModel = readSiteModel();
        Configuration cfg = getTemplateConfiguration();
        generateIndex(targetDirectory, siteModel, cfg);
        siteModel.getProduct().stream().forEach(product ->
generateProduct(targetDirectory, cfg, product));
    }

    private static void generateIndex(File targetDirectory, SiteType
siteModel, Configuration cfg) {
        try(Writer out = new FileWriter(new File(targetDirectory,
"index.html"))) {
            Template template = cfg.getTemplate("index-
template.html");
            template.process(siteModel, out);
        } catch (Exception e) {
            throw new IllegalArgumentException(e);
        }
    }

    private static void generateProduct(File targetDirectory,
Configuration cfg, ProductType product) {
        try(Writer out = new FileWriter(new File(targetDirectory,
product.getId() + ".html"))) {
            Template prodTemplate = cfg.getTemplate("product-
template.html");
            prodTemplate.process(product, out);
        } catch (Exception e) {
            throw new IllegalArgumentException(e);
        }
    }
}
```

```
    }  
    ...  
}
```

Чтобы на компьютере у барышни не приходилось ничего устанавливать кроме JDK, в проект добавил [mvn-classloader-1.8.jar](#), который в первый раз загружает из центрального maven репозитория maven-embedder и запускает сборку проекта и генерацию сайта с параметрами «clean package» и «scm-publish:publish-scm» для публикации сайта на GitHub Pages. Запускает генерацию сайта [build.cmd](#) для систем на windows и [build.sh](#) для linux. Это сильно упрощает сборку проекта и очень похоже на Gradle Wrapper/Maven Wrapper. Библиотека для maven требует только наличие JVM и подключения к интернет с доступом к центральному репозитарию. Случай изолированных сетей здесь рассматривать не будем, но даже в этом случае работают настройки зеркал и прокси из ~/.m2/settings.xml. Про всю магию этой библиотеки недавно рассказывал в «[Модуляризация в JavaSE без OSGI и Jigsaw](#)».

Пару слов про процесс обучения. Рассказывать о системе контроля версий и командах в консоли для Git мне было сложно. Не то что это невыполнимо, но не программиста это пугает. Чтобы она не ошибалась в процессе публикации сайта, я начал искать возможность автоматизировать весь процесс. Чтобы сгенерировать и опубликовать сайт можно было в один клик. Поискав как же легко публиковать сайт в github, обнаружил maven-scm-publish-plugin плагин. Добавил его в maven сборку и сконфигурировал scmPublish.pubScmUrl на адрес репозитория.



Отвлекусь от технологий и вспомню про женский праздник. Все реже можно видеть современных девушек за шитьем и рукоделием. А тут была возможность понаблюдать за работой дизайнера и как она шила:



Узнал про новую для себя область дизайна текстиля в интерьере. И даже не задумывался что существует столько видов штор, тканей и карнизов.

Еще до истории с сайтом смотрел, как она мастерила вместе с мамой подарок на День Рождения друга. Это были забавные чехлы на подушки с котенком и рыбкой, с собачкой.



Ээх, жаль что мне так давно не дарили handmade подарки, даже немного позавидовал.

Было интересно узнать про карнизы с электроприводом для «умного дома» и про опыт их установки. Хочу поэкспериментировать с электроприводом штор, прикрутить их [к своему контроллеру](#) на основе STM32 и автоматизировать открытие штор по датчику освещенности или управлять шторами удаленно с телефона. Разобраться с протоколом, подключить свой

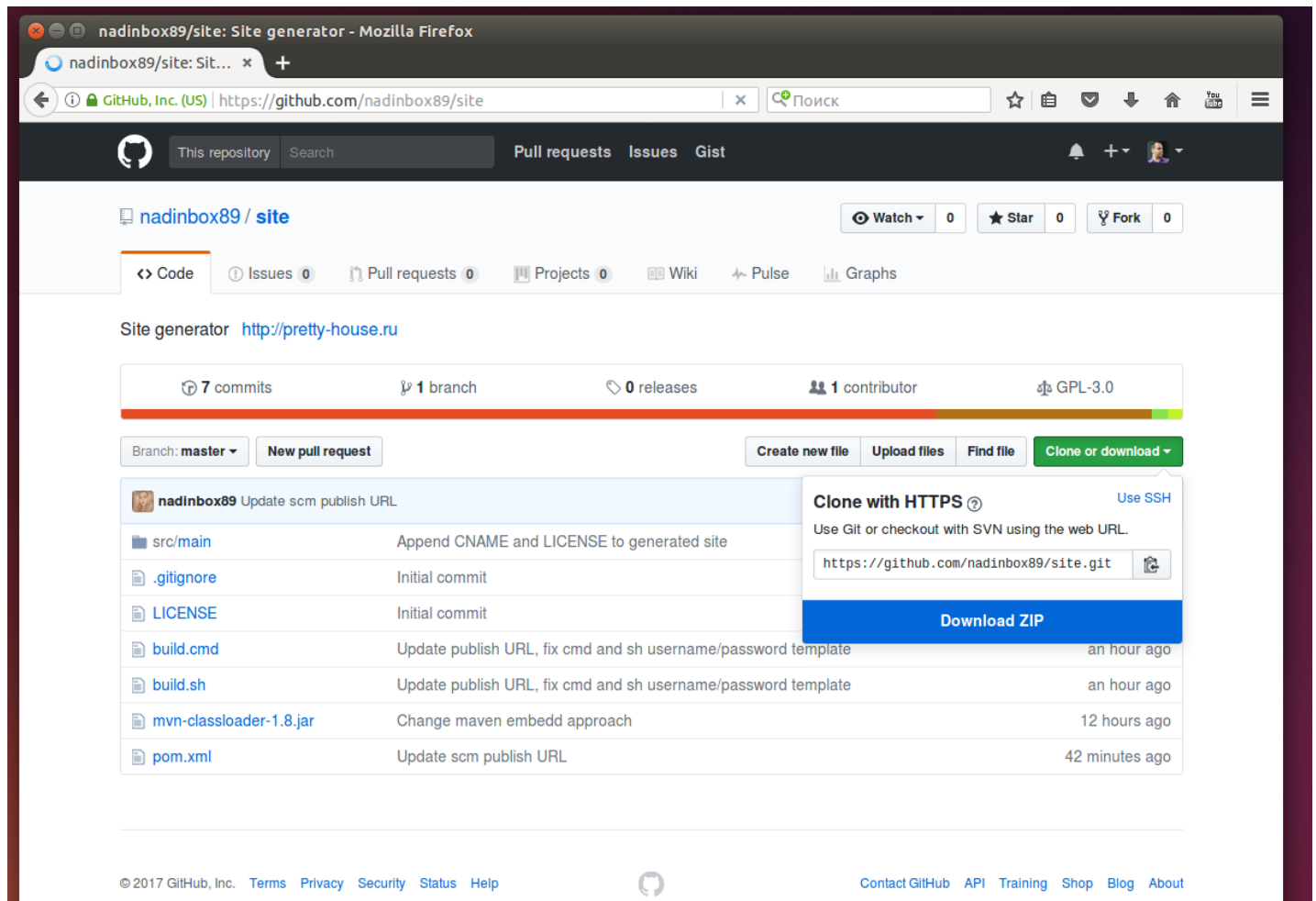


контроллер красиво и без синей изоленты! Но это уже будет, видимо, после того как разберусь с шаблонами jekyll. До этого видел подобные автоматические рулонные шторы только в Nordstar Tower.

Про то как обновлять контент... Первым делом надо выполнить команду

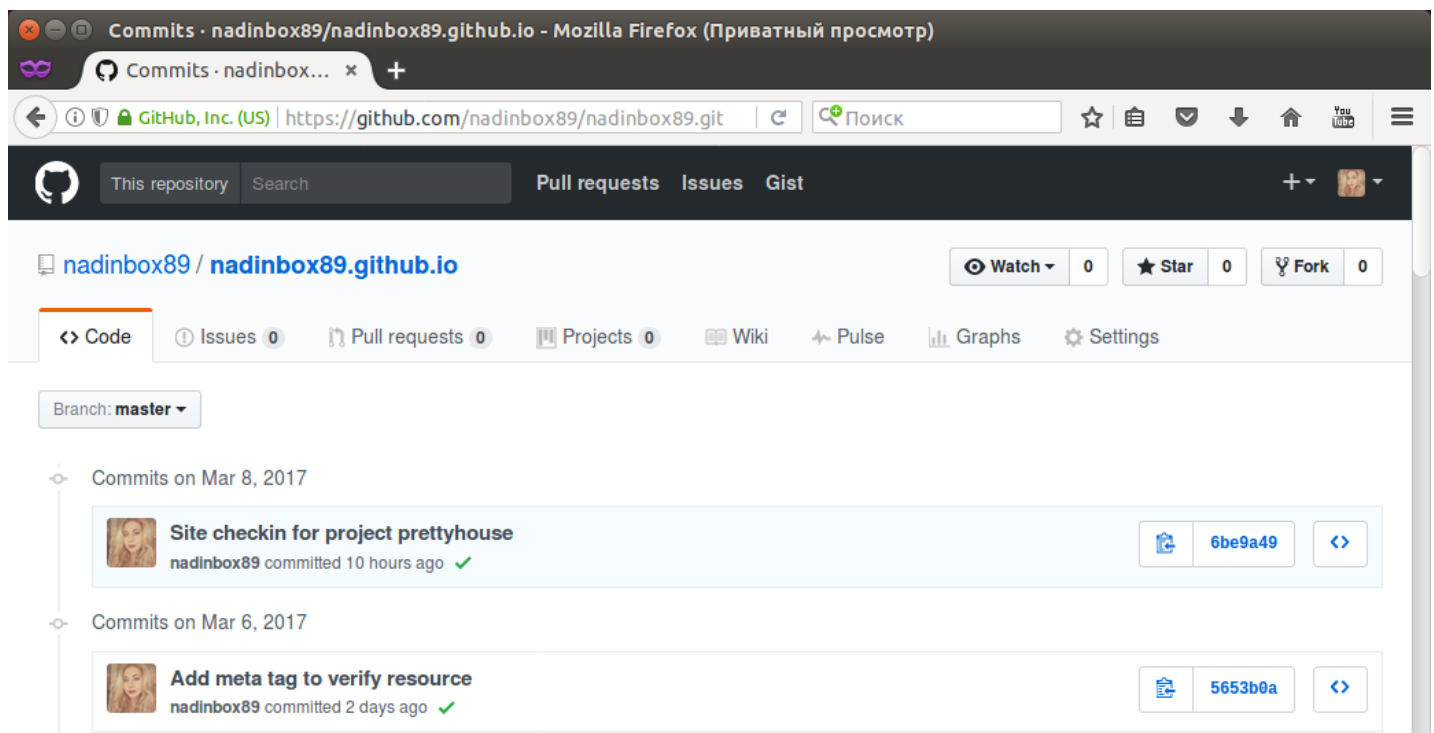
```
git clone https://github.com/nadinbox89/site.git
```

или скачать zip файл проекта с гитхаба:



Дизайнер копирует обработанные изображения для портфолио в

директорию `src/main/resources/img` и добавляет запись в раздел `<product...>` файла `src/main/resources/site.xml`. После этого запускает с помощью ярлыка скрипт `build.cmd`, который создает сайт из данных по шаблонам и публикует его на GitHub Pages в репозиторий `nadinbox89.github.io`. С небольшой задержкой после `git push`, версия сайта обновляется по такому же адресу, как и название репозитория на GitHub — `nadinbox89.github.io`. Если используете в репозитории запись домена в CNAME, то произойдет автоматическое перенаправление на этот адрес. После обновления сайта, в репозитории с именем `nadinbox89.github.io` появляется commit от плагина публикации:



Хорошо бы было спрятать от нее работу с XML совсем, но писать UI для генератора пока нет желания.

Сайт-визитку украсили доменным именем. Взять домен можно почти у любого регистратора за пару сотен рублей. После этого нужно добавить в корень вашего гитхаб репозитория сайта файл

CNAME с именем домена. Отлично, если у регистратора доменов есть бесплатные DNS сервера. Их нужно сконфигурировать для GitHub Pages следующим образом:

@ A 192.30.252.153

@ A 192.30.252.154

www CNAME \*ваш\_github\_аккаунт\*.github.io.

Где 192.30.252.153, 192.30.252.154 — IP адреса гитхаба. Про настройку доменного имени хорошо описано на [stackoverflow](#) и в помощи [GitHub](#). После этого проверил сервисом [mobile-friendly](#) и, в целом, сайт неплохо работает с мобильными браузерами.

The screenshot shows the Google Search Console interface in a Mozilla Firefox browser. The page title is "Проверка оптимизации для мобильных – Google Search Console - Mozilla Firefox". The address bar shows the URL "https://search.google.com/search-console/mobile-friendly?id=bkiVa6gXww-UM\_". The main heading is "Проверка оптимизации для мобильных". Below it, the URL "http://pretty-house.ru/" is displayed. A green message states: "Проверено 8 мар. 2017 г. в 11:48. Страница оптимизирована для мобильных устройств. Эта страница оптимизирована для мобильных устройств." There is a button "ДОБАВИТЬ В GOOGLE". A section titled "Дополнительная информация" contains three links: "Отчет об удобстве для мобильных", "Подробнее о страницах, оптимизированных для мобильных устройств...", and "Перейти на форум". On the right, a mobile preview of the website "Студия дизайна домашнего текстиля" is shown, featuring a photo of green patterned curtains. The preview includes a title, a description, a photo, and a text block about the studio's services. At the bottom left, there are links for "Политика конфиденциальности" and "Условия использования". At the bottom right, there is a share icon.

Проверка оптимизации для мобильных – Google Search Console - Mozilla Firefox

Проверка оптимиз... x +

https://search.google.com/search-console/mobile-friendly?id=bkiVa6gXww-UM\_ Поиск

Проверка оптимизации для мобильных

http://pretty-house.ru/

Проверено 8 мар. 2017 г. в 11:48

**Страница оптимизирована для мобильных устройств**

Эта страница оптимизирована для мобильных устройств

ДОБАВИТЬ В GOOGLE

Дополнительная информация

- Отчет об удобстве для мобильных
- Подробнее о страницах, оптимизированных для мобильных устройств...
- Перейти на форум

Студия дизайна домашнего текстиля

Примеры работ можете просмотреть, выбрав название раздела со значком ➔ в изображениях:

Портьеры, тюль

Мы, как мастера своего дела с многолетним стажем и солидным портфолио выполненных проектов, поможем вам с декорированием оконных проемов в жилом помещении или офисе. Можем выполнить весь спектр работ от замера и индивидуального проекта, до подбора материалов, пошива и установки. Выполним проект "под ключ" и подберем ткани, карнизы или жалюзи именно под ваш интерьер, ваши вкусы и ваш стиль.

Политика конфиденциальности Условия использования



Надеюсь, что мое участие в создании сайта закончилось и поддержка будет в таком же режиме, как в [старом добром видео](#)

## Заключение

Для хостинга простого статичного сайта-визитки отлично подходит GitHub Pages и все затраты — это время на верстку и покупка доменного имени за пару сотен рублей. Алгоритм не сложный и описан в этой статье:

1. Регистрация аккаунта на github
2. Создание репозитория `*ваш_github_аккаунт*.github.io`
3. Сохранение статики (html, css, изображений) в репозиторий `*ваш_github_аккаунт*.github.io`
4. Регистрация домена и создание записей DNS сервера указывающих на адрес из 3 пункта
5. Добавить файл CNAME в корень репозитория с именем вашего домена

Генерировать html можно как с помощью встроенного в GitHub шаблонного `jeekyll`, так и с помощью `java` программы и шаблонов на `FreeMarker`, что с моим опытом показалось проще. Так же удобно публиковать сайт на GitHub сразу из `maven` сборки с помощью `maven-scm-publish-plugin`. Требуется только настроить логин/пароль в скрипте, чтобы плагин для публикации смог выполнить `git push`. Мой подарок к 8 марта удался!



+12



Поделиться:



Сохранить:



Комментарии (56)

## Похожие публикации

### Sphinx-документация на GitHub Pages

yanchenko • 20 мая 2013 в 03:37

2

### GitHub Pages переезжают на github.io

hell0w0rd • 6 апреля 2013 в 02:39

12

### Natural Docs + GitHub Pages = online auto-documentation на халяву (почти)

meettya • 6 ноября 2011 в 21:09

31

## Популярное за сутки

## Яндекс открывает Алису для всех разработчиков. Платформа Яндекс.Диалоги (бета)

BarakAdama • вчера в 10:52

69

## Почему следует игнорировать истории основателей успешных стартапов

ПЕРЕВОД

m1rko • вчера в 10:44

20

## Как получить телефон (почти) любой красоты в Москве, или интересная особенность MT\_FREE

ИЗ ПЕСОЧНИЦЫ

sab404 • вчера в 20:27

24

## Java и Project Reactor

zealot\_and\_frenzy • вчера в 10:56

10

## Пользовательские агрегатные и оконные функции в PostgreSQL и Oracle

erogov • вчера в 12:46

6

## Лучшее на Geektimes

## Как фермеры Дикого Запада организовали телефонную сеть на колючей проволоке

NAGru • вчера в 10:10

31

## Энтузиаст сделал новую материнскую плату для ThinkPad X200s

49

alizar • вчера в 15:32

## Кто-то посылает секс-игрушки с Amazon незнакомцам. Amazon не знает, как их остановить

85

Pochtoycom • вчера в 13:06

## Илон Маск продолжает убеждать в необходимости создания колонии людей на Марсе

139

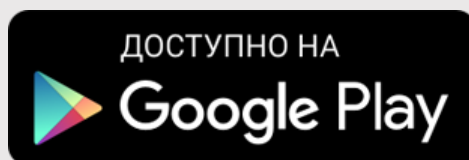
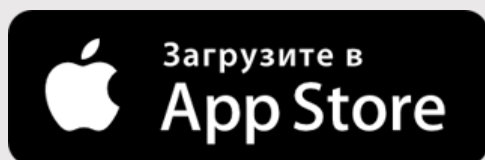
marks • вчера в 14:19

## Дела шпионские (часть 1)

16

TashaFridrih • вчера в 13:16

Мобильное приложение



Полная версия

2006 – 2018 © TM