

2019年8月19日

株式会社インプレスR&D

<https://nextpublishing.jp/>

Netlify を使ってらくらく Web サイト運用！
『ゼロから始める Netlify』発行
技術の泉シリーズ、8月の新刊

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『ゼロから始める Netlify』（著者：渡邊 達明、藤澤 伸、姫野 佑介）を発行いたします。

最新の知見を発信する『技術の泉シリーズ』は、「技術書典」をはじめとした各種即売会や、勉強会・LT 会などで頒布された技術同人誌を底本とした商業書籍を刊行し、技術同人誌の普及と発展に貢献することを目指します。

『ゼロから始めるNetlify』

<https://nextpublishing.jp/isbn/9784844396895>



著者：渡邊 達明、藤澤 伸、姫野 佑介

小売希望価格：電子書籍版 1600 円(税別)／印刷書籍版 1800 円(税別)

電子書籍版フォーマット：EPUB3／Kindle Format8

印刷書籍版仕様：B5 判／カラー／本文74ページ

ISBN：978-4-8443-9689-5

発行：インプレス R&D

<<発行主旨・内容紹介>>

Netlify は、PHP などを利用せずに、HTML などの静的コンテンツのみで構成された Web サイトを閲覧できる形で、運用・配信するための Web サービスです。

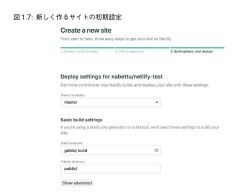
GitHub など管理しているリポジトリから自動的にデプロイを行える他に、フォームや CI 機能など静的サイトを運用する上で便利な機能が豊富に揃っています。

本書は、Netlify の豊富な機能を詳しく紹介し、活用方法を提案しています。
(本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。)

本書の対象読者

- 普段 Web サイトを運用していて、もっと楽にできる方法を探している人
- いつも FTP ソフトでいちいちレンタルサーバーにアップロードして消耗している人
- AWS S3 や GitHub Pages をいつも使っているけれど、今ひとつかゆいところに手が届かないと感じている人
- Netlify を使ったことはあるけど、機能が多くて何ができるのか分かってない人
- Netlify の便利そうな機能を、実際にどう使うかサンプルが見たい人

基本的な登録方法や機能紹介、設定方法を解説



この設定は、基本的にリポジトリ内の設定ファイルから自動取得されます。package.json の script に build コマンドがあれば、そこから自動的に取得して設定します。テンプレートなどを利用する際には、なにを設定をなくしても進めるだけでOKです。

「Deploy site」を押下してサイトを公開しましょう。サイト名を変更しなければさほど同じ手順で変更が可能です。

1.4.1 SPA で HistoryAPI (pushState) を利用したサイトを公開する際の設定

React.js や Vue.js などの SPA (シングルページアプリケーション) ライブラリを利用して複数ページあるサイトを公開し、ルーティングに HistoryAPI (pushState) を利用している際、トップページ以外に遷移後に再読み込みをすると 404 エラーとなってしまう場合があります。

これは通常のサイトのページ遷移と違い、SPA ライブラリが遷移時にサーバーのファイルを読み込まずに JavaScript を使ってページの内容を更新しているためです。ブラウザの URL は HistoryAPI を利用しているため切り替わってはいますが、トップ以外のページに html ファイルが無いと再読み込みで 404 エラーとなります。(バグがはからぬハッシュルーティングを利用している場合は問題ありません)

この問題は、ファイルが見つからなかった場合でもルートに配置してある index.html を読み込むように設定すれば解消されます。

そのために netlify.toml という設定ファイルをリポジトリルートに配置します。こちらは Netlify でホスティングを行う際にリダイレクトなどの設定をまとめておくための設定ファイルです。次の記述を追加しておきましょう。

```
netlify.toml
[[redirects]]
  from = "/*"
  to = "/index.html"
  status = 200
```

また、こちらの設定は、.netlify/_redirects にリダイレクト設定を記載する方法もあります。記述が多くなりファイルを分割したい場合などにはこちらも利用できます。

1.5 Hosting の速度について

この章で、無事サイトを公開できたかと思いますが、ここで気になるのが、「他のホスティングサービスと比べて速度はどうなの？」というところです。

表示スピードに関しては、SEO 対策としても近年取り上げられるようになってきたため、とても気になることです。今回は、次の4つのホスティング方法で同じソースをデプロイし、Chrome DevTools の「Network」タブで index.html を転送する速度を調べてみました。

- AWS (S3+CloudFront)
- Firebase Hosting
- GitHub Pages
- Netlify

※ AWS は、ほかサイトと同程度にするため、HTTP/2 通信と gzip 圧縮によるキャッシュ配置に対応させて計測しています。

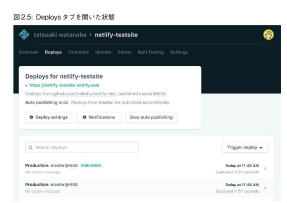
結果は図のようになりました。今回試した4つのホスティング方法では、どれも数msほどの差です。サイトアクセス時に「読み込みが遅いな」という原因として挙げることはすくなくないと考えてよいでしょう。

図 1.8: index.html のレスポンス送達速度

項目	AWS (S3+CloudFront)	Firebase Hosting	GitHub Pages	Netlify
Resource Scheduling				
Queueing	337ms	307ms	274ms	
Connection Start				
Socket	2.84ms	2.23ms	4.14ms	5.46ms
Request / Response				
Request sent	1.41ms	3.25ms	6.29ms	0.74ms
Waiting (TTFB)	163.05ms	169.78ms	171.87ms	168.68ms
Connect Download	49.22ms	48.22ms	47.88ms	47.76ms
Total	618.02ms	623.59ms	625.65ms	623.56ms

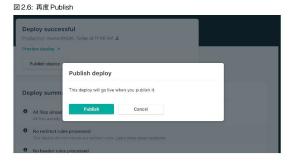
11 | 第1章 登録方法や基本機能の紹介 第1章 登録方法や基本機能の紹介 | 15

独自ドメインを設定した Web サイトの公開方法を紹介



ここでは過去のデプロイのログが表示されています。GitHub リポジトリの commit と連携して、サイトがデプロイされた記録が残っています。その中で「PUBLISHED」がついたバージョンが現在サイトに表示されているバージョンです。

原たいバージョンを選択してクリックしてください。するとその Deploy のビルドログが閲覧できます。そのまま「Publish deploy」を押すと出てくるモーダル上の「Publish」ボタンを押すと公開されているサイトがそのバージョンに切り替わります。



こちらは以前のビルドのデータを Netlify が保持してくれているので、再度ビルドを走らせずに同時に戻すことが可能です。これでバグの無かったバージョンを公開してから、安心してデバッグしていきましょう。

また注意点として、連携リポジトリにまた新しく push すると最新のコードが自動で公開されます。もし確認してからデプロイしたい場合は Deploy タブを開いた際の「Stop auto publishing」を押して自動公開を止めておきましょう。

ビルド周りの機能や設定の説明は以上です。細かいながらも便利な機能が関わっていることが分

かっていただけたと思います。

22 | 第2章 ビルド周りの機能や設定 第2章 ビルド周りの機能や設定 | 23

Netlify の機能の一つであるフォームの設置方法を紹介

リスト 4.5: editorid_workflow

```
publish_mode: editorial_workflow
```

この機能をONにすることで、記事の作成時にはそのまま公開されず、いったん下書きの状態
保存・その後別のユーザーからのレビューを採んだ後 (GitHub 上で Pull Request が作成され
ています)、公開するというような運用も可能です。

図 4.15: 投稿画面に「Save」ボタンが現れる

なおこの機能は内部的には、新規投稿をいきなりデフォルトブランチにコミットするのではなく
別のブランチにコミットすることで、「コミットはするが公開はしない」「レビュー後に公開でき
る」という状況をつくりだしています。

さらにビルドの章で触れた「ブランチごと確認用のサイトを作る」手法と組み合わせれば、この
下書き状態の投稿についても Preview Deploy の別 URL から確認することができますので、最終的
な見栄えをプレビューしながら記事制作することも可能です。

4.5.2 投稿内容のカスタマイズ

この章で紹介した config.yml を使った場合、記事投稿時に入力できるフィールドは、Title / Publish
Date / Featured Image / Category / Body の5つでしたが、こちらはもちろんカスタマイズ可能
です。

- <https://www.netlifycms.org/docs/add-to-your-site/#collections>
投稿に含まれる入力フィールドを追加・削除したり、それぞれのフィールドを入力する際に使用す
るウィジェット (短いテキストボックス・マークダウンが着けるリッチエディター・チェックボッ
クス・プルダウン etc...) も選択することができます。

図 4.16: NetlifyCMS に標準搭載されている、入力ウィジェット

Default widgets

Boolean	Color	Date/Time	File	Hidden	Image	List	Markdown
Number	Object	Relation	Select	String	Text		

Date

The date widget translates a date picker input to a date string. For saving date and time together, use the relation widget.

また WordPress でいうところの「カスタム投稿タイプ」のように、記事以外のコンテンツも入稿
できるような投稿の種類自体を増やすことも可能です。こちらを使えば、NetlifyCMS の用途はブログ
のようなサイトだけではなく、うまく組み合わせればコーポレートサイトや EC サイトのよう
なものも実現可能でしょう。

4.5.3 管理画面ユーザー数の拡張

複数人で CMS サイトの運用を行いたい場合、自分だけでなく他のユーザーも CMS に招待したく
なると思います。この際注意なのは、Netlify の無料プランでは **5 人までしか招待ユーザーを作成で
きない**ことです。それ以上のユーザーを作成したい場合は、「Identity Pro」プランに切り替える必
要があります。

図 4.17: Identity Pro プランの詳細

Upgrade to Identity Pro

Your team requires all features available immediately.

- **Active users:** Up to 5,000 per month.
All users are managed by you and are active when you connect them to your site.
- **Invite-only users:** Up to 100.
For sites not by the owner. Grants all users in your team full, regardless of sign-in status.
- **Granular control of OAuth integrations.**
You decide any and how you use the remaining apps when authorizing external sign-in providers.
- **Custom outgoing email address.**
Get personalized emails for invitations, notifications, and password reset emails.
- **Audit log:** 7-day retention.
History of all activity in your identity service.

Pro プランではその他にも、招待メールの送信元をカスタマイズしたり、ユーザーのアクティビ
ティログを閲覧できるようにしたり、さまざまな機能が追加されます。大規模なサイト構築を計画
している方は、ぜひ調べてみてください。

42 | 第 4 章 CMS をつくる

第 4 章 CMS をつくる | 43

<< 目次 >>

第 1 章 登録方法や基本機能の紹介

- 1.1 まずは登録してみよう
- 1.2 さっそくサイトを公開してみよう
- 1.3 公開 URL を変更する
- 1.4 GitHub と連携してみよう
- 1.5 Hosting の速度について

第 2 章 ビルド周りの機能や設定

- 2.1 ビルドって？
- 2.2 Git でのブランチごとにサイトを作る
- 2.3 ビルド時にビルドコマンドを実行するディレクトリを変更する
- 2.4 ビルド時に変数を設定する
- 2.5 Webhook でビルド&デプロイさせる
- 2.6 ファイル配信時の Netlify 側での前処理
- 2.7 デプロイが終わったら Slack 等に通知を送る
- 2.8 サイトをいったん前のバージョンに戻したい場合

第 3 章 独自ドメインを割り当ててみよう

- 3.1 Netlify Managed DNS とは
- 3.2 ドメイン購入を Netlify 上でやってみる
- 3.3 外部のドメイン購入サービスで買ったドメインを当ててみる
- 3.4 SSL 対応 (HTTPS 化) を行う

第 4 章 CMS をつくる

- 4.1 NetlifyCMS とは
- 4.2 「deploy to Netlify」で素早く試す

- 4.3 仕組みの全体像を知る
- 4.4 スクラッチでの構築
- 4.5 CMS をもっと便利にする
- 第5章 フォームの設置方法
 - 5.1 基本的な設置方法
 - 5.2 フォームをカスタマイズ
 - 5.3 メールで問い合わせを受け取る
 - 5.4 料金形態について
- 第6章 Split Testing
 - 6.1 利用手順
 - 6.2 機能
 - 6.3 注意事項
 - 6.4 ブランチ名をスクリプト側で取得する
 - 6.5 ブランチごとのデータを Google Analytics に送信する
- 第7章 Functions (AWS Lambda on Netlify)
 - 7.1 概要
 - 7.2 準備
 - 7.3 まずは、Hello World
 - 7.4 使用例
 - 7.5 netlify-lambda を活用する
- 第8章 Prerendering 機能を試す
 - 8.1 設定方法
 - 8.2 仕組み
 - 8.3 注意事項
 - 8.4 Prerendering されていない時は
 - 8.5 別の Prerendering サービスを利用したい
- 第9章 チーム機能や有料プランでできること
 - 9.1 チームのユーザー毎に役割を設定する
 - 9.2 特定のユーザーのみにアクセスを許可する
 - 9.3 有料プランになるとできること

<< 著者紹介 >>

渡邊 達明(わたなべ たつあき)

株式会社クリモ取締役副社長。1988 年宮城県生まれ。仙台高専専攻科を卒業後、富士通株式会社にて WindowsOS のカスタマイズ業務に従事する。

その後面白法人カヤックにて受託開発部門を経験後、プログラマーの妻と二人で株式会社クリモを設立。Web フロントエンドや React Native の受託開発や保育園問題の解決のためのメディアを運営。

「三度の飯よりものづくり」と言っていたら BMI が 17 になり健康診断で毎回ひっかかるのが悩み。一番好きな寿司ネタは「えんがわ」。

twitter: @nabettu

blog: <http://blog.nabettu.com/>

藤澤 伸(ふじさわ しん)

面白法人カヤック・クライアントワーク事業部技術部統括。1990 年生まれ。慶應義塾大学 SFC にてプログラミング・デザイン・作曲等々を学んだ後、新卒でカヤックに入社。

主には受託開発部門のフロントエンドエンジニアとして、時にはテクニカルディレクターとして、時にはサウンドクリエイターとして、時にはアイドルプロデューサーとして働く。一番好きな寿司ネタは「えんがわ」。

twitter: @fnobi

blog: <http://fnobi.com/>

姫野 佑介(ひめの ゆうすけ)

面白法人カヤックのWebエンジニア。1993年生まれ。カヤックに新卒で入社。フロントエンドエンジニアを主軸としながらバックエンドエンジニアにまで幅を広げて受託開発部門で働く。

大体ネックウォーマーを着用している。好きな寿司ネタは「サーモン」。

twitter: @hyme_

blog: <https://hyme.site/>

<< 販売ストア >>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple Books、紀伊國屋書店 Kinoppy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

※ 各ストアでの販売は準備が整いしだい開始されます。

※ 全国の一般書店からもご注文いただけます。

【インプレス R&D】 <https://nextpublishing.jp/>

株式会社インプレスR&D(本社:東京都千代田区、代表取締役社長:井芹昌信)は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishing を使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知の流通を目指しています。

【インプレスグループ】 <https://www.impressholdings.com/>

株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:唐島夏生、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「旅・鉄道」「学術・理工学」を主要テーマに専門性の高いメディア&サービスおよびソリューション事業を展開しています。さらに、コンテンツビジネスのプラットフォーム開発・運営も手がけています。

【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

TEL 03-6837-4820

電子メール: np-info@impress.co.jp