

Quaint-wa レポジトリの作成手順書

記録用。

① 新規レポジトリ作成

(MIT ライセンスをつけておくべきだった…現在 api, docs だけついている)

参考：[GitHub が公開してるオープンソースガイドライン、まとめてみた。](#)

② git clone で複製

参考：[【Git,GitHub】リモートリポジトリを複製する方法](#)

```
管理者ユーザ - @pasokon MINGW64 ~
$ git clone --bare https://github.com/hibiya-itchief/quaint-app.git
Cloning into bare repository 'quaint-app.git'...
remote: Enumerating objects: 3973, done.
remote: Counting objects: 100% (2096/2096), done.
remote: Compressing objects: 100% (910/910), done.
remote: Total 3973 (delta 1379), reused 1682 (delta 1093), pack-reused 1877
Receiving objects: 100% (3973/3973), 21.07 MiB | 1.88 MiB/s, done.
Resolving deltas: 100% (2163/2163), done.

管理者ユーザ - @pasokon MINGW64 ~
$ cd quaint-app.git

管理者ユーザ - @pasokon MINGW64 ~/quaint-app.git (BARE:develop)
$ git push --mirror https://github.com/hibiya-itchief/quaint-wa.git
Enumerating objects: 3973, done.
Counting objects: 100% (3973/3973), done.
Delta compression using up to 4 threads
Compressing objects: 100% (1569/1569), done.
Writing objects: 100% (3973/3973), 21.07 MiB | 1.13 MiB/s, done.
Total 3973 (delta 2163), reused 3973 (delta 2163), pack-reused 0
remote: Resolving deltas: 100% (2163/2163), done.
To https://github.com/hibiya-itchief/quaint-wa.git
 * [new branch]      #177-fix-ticketpage -> #177-fix-ticketpage
 * [new branch]      0710お気に入り機能修正 -> 0710お気に入り機能修正
 * [new branch]      189-adminが新規団体を作れる画面 -> 189-adminが新規団体を作れる画面
 * [new branch]      214-post -> 214-post
```

② pages 作成とその他設定する、デプロイフック URL 発行 (およびバグ対処、詳しくは issue #3)

The screenshot shows the Cloudflare dashboard interface for a project named "IT Chief | Tokyo Metr...". The left sidebar contains navigation options such as "Web サイト", "Discover", "ドメイン登録", "分析とログ", "セキュリティ センター", "Trace", "Turnstile", "Zero Trust", "Area 1", "Workers & Pages", and "AI". The main content area is titled "ビルドとデプロイのセットアップ" (Build and Deploy Setup) for the project "hibiya-itchief/quaint-wa".

At the top of the main area, there are three steps: 1. リポジトリの選択 (Repository selection), 2. ビルドとデプロイのセットアップ (Build and Deploy Setup), and 3. サイトをデプロイ (Deploy site). The current step is "ビルドとデプロイのセットアップ".

The setup page includes the following fields and options:

- プロジェクト名** (Project name): Input field containing "quaint-wa". Below it, a note states: "プロジェクトはquaint-wa.pages.devにデプロイされます。"
- プロダクション ブランチ** (Production branch): Dropdown menu set to "main". Below it, a note states: "このブランチへのプッシュにより、本番環境へのデプロイが自動的に開始されます。他のすべてのブランチへのプッシュにより、プレビュー環境へのデプロイが開始されます。"
- ビルドの設定** (Build settings): Includes a link to "ビルドの構成" (Build configuration). A note states: "プロジェクトで静的サイトジェネレーターやビルドツールを使用している場合は、Cloudflare用のビルド手順を設定します。"
- フレームワークプリセット** (Framework preset): Dropdown menu set to "Nuxt.js". Below it, a note states: "推奨設定を満たすフレームワークを選択します。"
- ビルドコマンド** (Build command): Input field containing "yarn && yarn run generate:prod". Below it, an example is given: "例: npm run build".
- ビルド出力ディレクトリ** (Build output directory): Input field containing "/ dist". Below it, an example is given: "例: dist".
- ルートディレクトリ (アドバンスド)** (Advanced root directory): Expandable section.
- 環境変数 (アドバンスド)** (Advanced environment variables): A note states: "本番環境とプレビュー環境のビルド時に使用する変数を追加します。各環境の変数は、後で個別に変更することができます。"
- 変数名** (Variable name): Input field containing "NODE_VERSION".
- 値** (Value): Input field containing "16.16.0".
- + 変数を追加する** (Add variable button): Button to add more variables.

At the bottom of the page, there are two buttons: "← リポジトリを変更" (Change repository) and "保存してデプロイする" (Save and Deploy).

③ Azure 内の「ログインできるサイト」に quaint-wa.pages.dev を追加
(アプリの追加→全てのアプリ→quaint-app→認証)

④ GitHub、レポジトリの設定を同じにする

※Environment 名だけ「quaint-app」に変更

のちにフロントと API 合わさった 1 レポジトリにしたときに、quaint-app,quaint-api-dev,quaint-api-prod という Environment 構成になる。

⑤ cloudflare pages にデプロイする actions(workflow)を編集：

Environment の名を quaint-app に変更, url を quaint-wa.pages.dev に変更

参考：[GitHub の Environments ことはじめ](#)

⑥～

- issue の label を見比べて同じにする
- issue の引っ越し
- Dependabot(処遇未定)