

Deploy SvelteKit-Projekt auf Codeberg-Pages

Beispielprojekt

- Repository-URL: <https://codeberg.org/meisterk/REPONAME>
- git-Branch: pages
- Pages-URL: <https://meisterk.codeberg.page/REPONAME/>

Adapter installieren

```
npm i -D @sveltejs/adapter-static
```

svelte.config.js (REPONAME anpassen)

```
import adapter from '@sveltejs/adapter-static';
export default {
  kit: {
    adapter: adapter({
      pages: 'build',
      assets: 'build',
      fallback: undefined,
      precompress: false,
      strict: true,
      fallback: 'index.html' // may differ from host to host
    }),
    paths: {
      base: process.env.NODE_ENV === 'production' ? '/REPONAME' : '',
    }
  }
};
```

src/routes/+layout.js

```
export const ssr = false;
export const prerender = false;
```

Links in Komponenten (Komponente.svelte)

```
<script>
  import { base } from '$app/paths';
</script>

<nav>
  <a href="{base}/" >Home</a>
  <a href="{base}/about" >About</a>
</nav>
```

Public Repository auf Codeberg erstellen

REPONAME

Linux: deploy.sh (REPONAME anpassen)

```
#!/usr/bin/env sh

# Bei Fehler sofort abbrechen
set -e

# build-Verzeichnis komplett löschen
rm -rf build

# build (build-Verzeichnis erstellen)
npm run build

# komplett neues Repository in build
# mit Branch 'pages' erstellen
cd build
git init
git checkout -B pages
git add -A
git commit -m 'deploy'

# aktuelle Version hochladen
# alte Version wird gelöscht
git push --force -u 'https://codeberg.org/meisterk/REPONAME.git' pages

# wieder in altes Verzeichnis springen
cd -
```

Windows: deploy.cmd

```
REM build-Verzeichnis komplett löschen
rd /s /q "build"

REM build
call npm run build

REM komplett neues Repository in build
REM mit Branch 'pages' erstellen
cd build
git init
git checkout -B pages
git add -A
git commit -m 'deploy'

REM aktuelle Version hochladen
REM alte Version wird gelöscht
git push --force -u https://codeberg.org/meisterk/REPONAME.git pages

REM wieder zurückspringen
cd ..
```