**Oficina GitHub**

Segue abaixo 3 exemplos de projetos de Github que estão bem documentados e que usam boas práticas de GitHub Pages. Com eles você pode visualizar e entender melhor como colocar essas boas práticas em seus projetos do GitHub.

**Projetos bem documentados**

1. PiggyMetrics (sqshq/piggymetrics)  
   Link: [https://github.com/sqshq/piggymetrics](https://github.com/sqshq/piggymetrics?utm_source=chatgpt.com)
   * Por que é bom: README claro, explica o que o projeto faz (“Microservice Architecture with Spring Boot, Spring Cloud and Docker”) e dá contexto.
   * Estrutura relativamente limpa, licença MIT, fácil de compreender.
   * Útil como referência de “projeto tutorial” que mostra patterns de microserviços.
2. DevOpsMetrics (DeveloperMetrics/DevOpsMetrics)  
   Link: <https://github.com/DeveloperMetrics/DevOpsMetrics>
   * README bem estruturado: apresenta o “por que” do projeto (“extract and process high performing DevOps metrics”), o que ele oferece.
   * Inclusive apresenta links, estrutura mista “src / README / LICENSE” de forma limpa.
   * Boa para ver como documentar ferramentas utilitárias / bibliotecas.
3. readmine (mhucka/readmine)  
   Link: [https://github.com/mhucka/readmine](https://github.com/mhucka/readmine?utm_source=chatgpt.com)
   * Projeto minimalista ~ “template” de README para outros projetos. Isso já por si é interessante: está focado em boa prática de documentação.
   * Ele mostra a estrutura sugerida de um README (“Introduction, Installation, Quick start, Usage, Known issues …”) que é algo que você pode adotar nos seus próprios projetos.

**Projetos com boas práticas de GitHub Pages**

1. senli1073.github.io (template para site pessoal acadêmico)  
   Link: [https://github.com/senli1073/senli1073.github.io](https://github.com/senli1073/senli1073.github.io?utm_source=chatgpt.com)
   * Um exemplo de template “site pessoal” com GitHub Pages: estrutura clara, README explicativo (“Getting Start”, passos de fork/uso)
   * Suporta Markdown como conteúdo, inclusive fórmulas LaTeX, o que mostra maturidade.
   * Uso adequado de nome de repo para site (<username>.github.io) conforme boas práticas de Pages.
2. Hosting‑a‑Static‑Website‑with‑GitHub‑Pages (dasaradhi10/…)  
   Link: [https://github.com/dasaradhi10/Hosting-a-Static-Website-with-GitHub-Pages](https://github.com/dasaradhi10/Hosting-a-Static-Website-with-GitHub-Pages?utm_source=chatgpt.com)
   * Projeto simples que demonstra de modo didático como hospedar um website estático com GitHub Pages. README explica o fluxo passo-a-passo (“1. Created index.html… 2. Initialized repo … 3. Enabled GitHub Pages …”)
   * Boa para ver o básico e adotar para aplicar rapidamente sites estáticos.
3. skills/github‑pages (template oficial de GitHub para Pages)  
   Link: [https://github.com/skills/github-pages](https://github.com/skills/github-pages?utm_source=chatgpt.com) [GitHub](https://github.com/skills/github-pages?utm_source=chatgpt.com)
   * Um dos exemplos “oficiais”/template orientados por boas práticas: README explica o que é (“Create a site or blog from your GitHub repositories with GitHub Pages”)
   * Usa estrutura com LICENSE, README.md, .gitignore — mostra que boas práticas de repositório se aplicam para sites estáticos também.