

# Como renderizar o Bioquanti fora do Debian gráfico do Active 5

Tá sonhando nada, Zema — isso é **totalmente real**. 😊

Na prática, **RStudio** é **opcional**: ele só é uma “casca” confortável em cima de **Quarto + R + Git**. Se isso roda no terminal, você consegue um fluxo **mais rápido e mais confiável** no Active 5.

## O fluxo “Terminal-only” (Quarto + Git + Netlify)

### 1) Render do site (na raiz do projeto **pt/**)

```
cd ~/Dropbox/pesq/producao/bioquanti/pt
quarto render
```

### 2) Ver rápido o resultado local (sem preview pesado)

```
python3 -m http.server 8000 --directory ../_site
```

Aí abre no navegador: <http://localhost:8000/pt/>

*(Isso costuma ser bem mais leve do que **quarto preview** em máquinas fracas.)*

### 3) Commit + push

```
cd ~/Dropbox/pesq/producao/bioquanti
git status
git add -A
git commit -m "Atualiza Bioquanti"
git pull --no-rebase origin main
git push origin main
```

👉 Daí o **Netlify** compila/deploya sozinho (se estiver configurado pra build). Você fica só no “render local pra conferir” + “push”.

---

**E dá pra renderizar sem RStudio? Sim. Sem R? Depende.**

- Se suas páginas têm **chunks R** (você mostrou que têm), você precisa de **R disponível no ambiente** que vai renderizar.
  - Mas o **render** em si é do Quarto (CLI), não do RStudio.
- 

## Termux / Android “nativo”: dá pra chegar perto

Duas opções bem reais:

### Opção A — Termux rodando Quarto + R

Funciona, mas é o caminho mais chatinho (instalar R no Termux, ajustar libs).  
Se você já tem Debian/proot redondo, talvez não valha mexer.

### Opção B — Debian no proot (o seu) + terminal

É o “sweet spot”: você já tem o stack e só troca a interface (RStudio → terminal).

---

## O truque que vai te dar MUITA velocidade no Active 5

Quando você não precisa reconstruir tudo:

```
cd ~/Dropbox/pesq/producao/bioquanti/pt
quarto render nivel/superior/hp50/Site1/Site1.qmd
```

Isso renderiza **só a página**, sem o site inteiro.

---

## Quer deixar isso “um botão só”?

Cria um script `render_pt.sh` no root `bioquanti/`:

```
#!/usr/bin/env bash
set -e

cd "$(dirname "$0")/pt"
quarto render
echo "OK: render concluído em ../_site/pt"
```

E roda:

```
bash render_pt.sh
```

Se quiser eu te faço também um `deploy.sh` que faz render + add/commit + pull --no-rebase + push (com mensagens bonitinhas e proteção contra conflito).

Kekulé ficaria orgulhoso — mas aqui é mais “engenharia de fluxo de trabalho” do que sonho

