Perfeito! Agora que você esclareceu que a estrutura é:

java

CopiarEditar

fiberguardian/ ← repositório Git (.git)

├── backend/ ← contém o .gitignore

│ ├── src/

│ ├── .gitignore ← está aqui

│ └── ...

├── frontend/

│ └── ...

└── .git/ ← repositório Git está aqui (na raiz)

**⚠️ Então atenção: .gitignore dentro de backend/ não controla nada fora de backend/, nem o diretório frontend/, e não é o ponto ideal para gerenciar ignore globais.**

**✅ Resposta objetiva:**

**Se o seu .gitignore está dentro de backend/**, então a linha:

gitignore

CopiarEditar

src/main/resources/cookies.txt

é válida e **funciona corretamente apenas para a estrutura interna a backend/** — ou seja, só ignora arquivos/diretórios **dentro de backend/**.

**✅ Melhor prática recomendada para seu projeto:**

📌 **Mover o .gitignore para a raiz do projeto** (fiberguardian/) e adaptar os caminhos:

**Novo caminho:**

bash

CopiarEditar

fiberguardian/.gitignore

**Conteúdo ajustado:**

gitignore

CopiarEditar

# Ignorar arquivos sensíveis do backend

backend/src/main/resources/cookies.txt

# Ignorar pastas de build

backend/target/

frontend/dist/

# IntelliJ

.idea/

\*.iml

# VSCode

.vscode/

# NetBeans

nbproject/

nbbuild/

dist/

# Maven Wrapper

.mvn/wrapper/maven-wrapper.jar

🔄 Assim, você **centraliza as regras de ignore** e passa a controlar tudo no projeto — backend, frontend, IDEs, builds etc. — com um único .gitignore.

**✅ Conclusão:**

* Sim, **src/main/resources/cookies.txt** funciona no .gitignore dentro de backend/, desde que esse caminho exista dentro de backend/.
* **Mas** se você quiser ignorar outros diretórios (frontend/, IDEs, etc.), **o ideal é mover .gitignore para a raiz do projeto (fiberguardian/)** e ajustar os caminhos.

Se quiser, posso gerar o conteúdo completo do .gitignore ideal para a sua estrutura, incluindo boas práticas para projetos Java com frontend e múltiplas IDEs. Deseja isso?