

Sistema de Arquivos Distribuídos Básico

BigFS (*simulação em um sistema distribuído em um único nó*)

Introdução ao *BigFS*

Objetivo

Simular operações básicas de arquivo em rede

Comandos

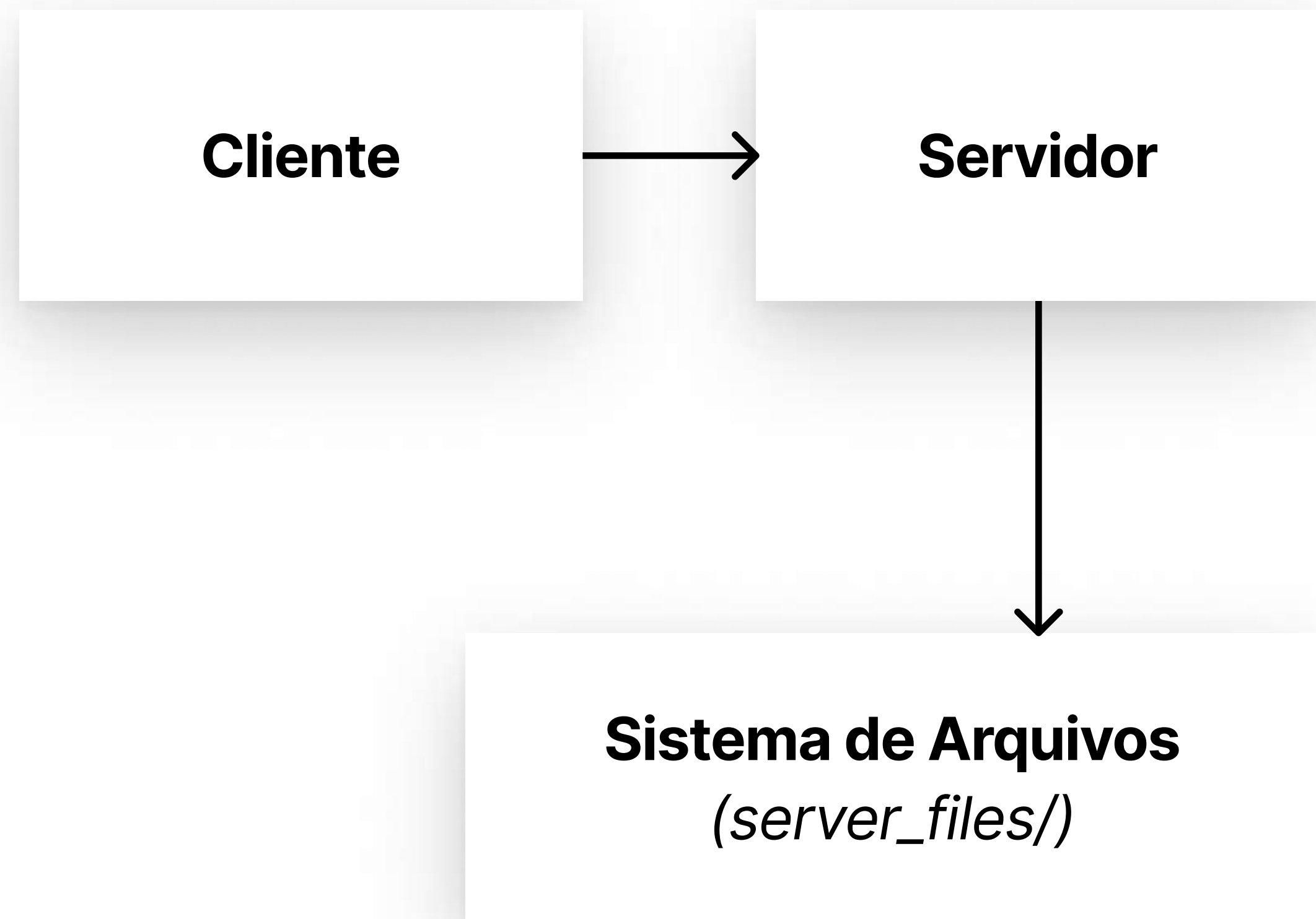
ls (listar) | **rm** (remover) | **cp** (copiar) | **get** (obter)

Tecnologias

Python, Sockets TCP, Threads

BigFS

Arquitetura do Sistema



Cliente → Interface de comandos via terminal

Servidor → Multi-threaded (múltiplos clientes)

Armazenamento → Diretório local (*server_files/*)

BigFS

Protocolos de Comunicação

Comandos

texto plano

Comando arg1 arg2 ...

Respostas

Sucesso

→

OK: [mensagem]

Erro

→

ERROR: [mensagem]

Dados binários

→

prefixo **FILE** [nome] [tamanho]\n

Client

"ls docs" -->

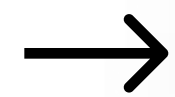
Server

<-- "arquivo1.txt\narquivo2.pdf\n"

BigFS

Protocolos de Comando:

ls



List (Listar)

Client

ls [path/file]

"FILE: [nome] Size: [tamanho]"

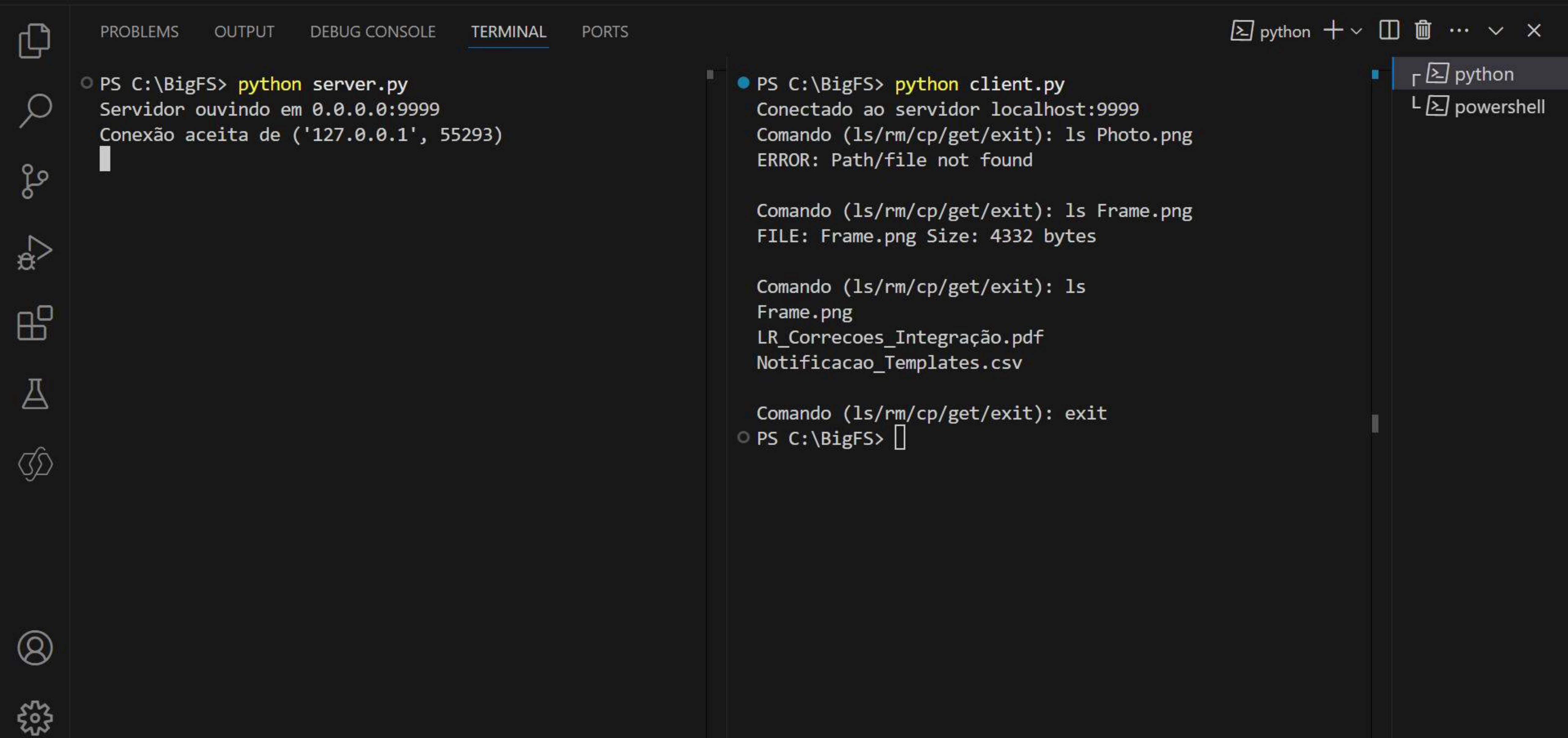
"ERROR: Path/file not found"

Server

Caminho Existe?

Arquivo ou Diretório?



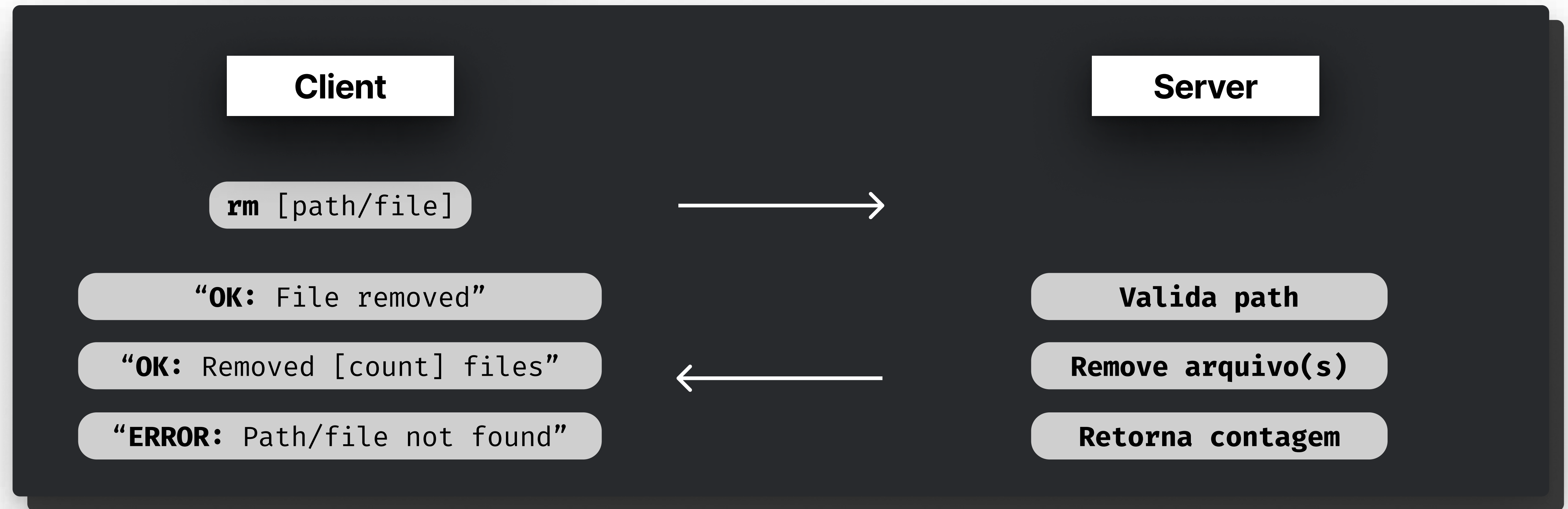


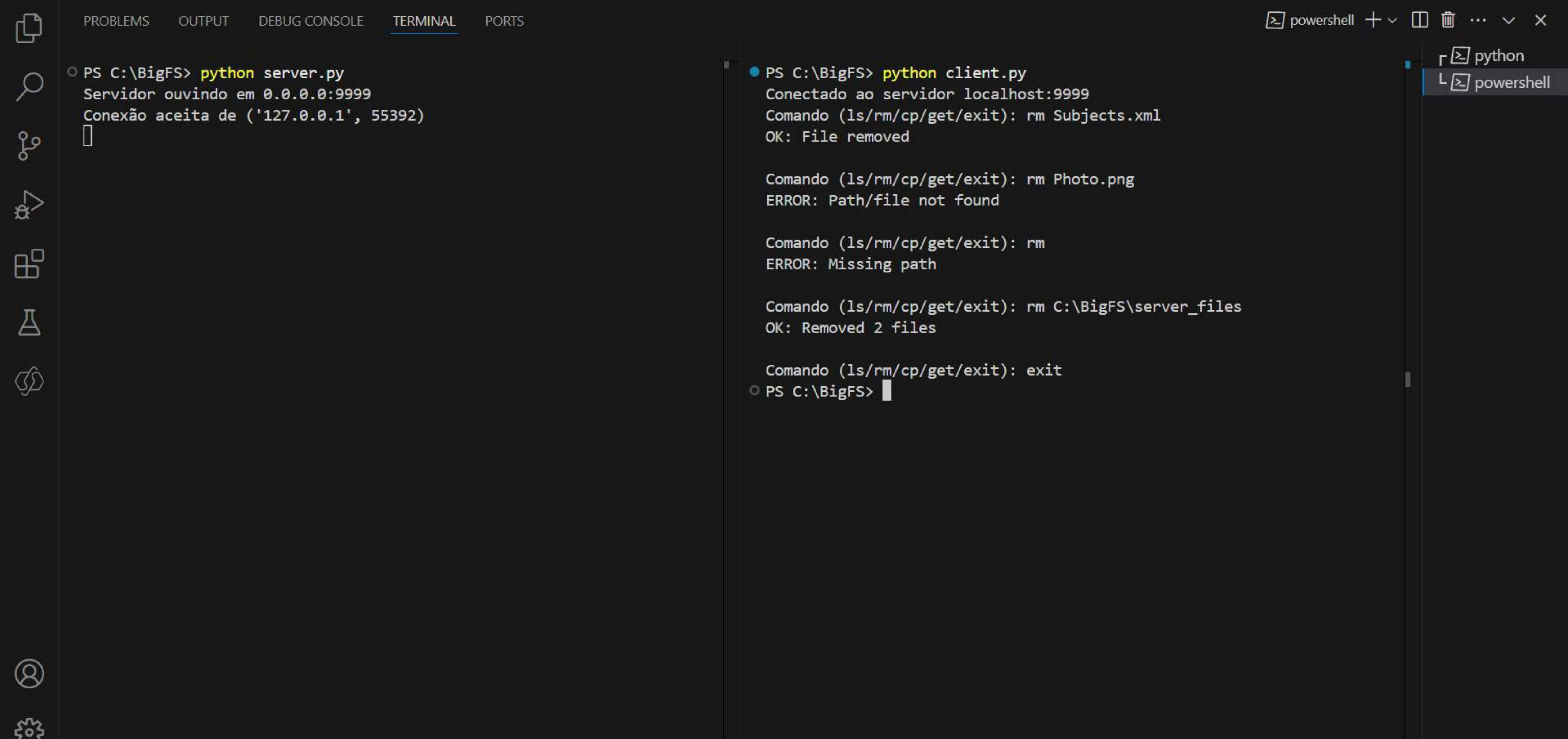
Demonstração ls

BigFS

Protocolos de Comando:

rm → **Remove (Remover)**



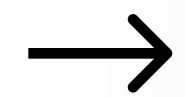


Demonstração rm

BigFS

Protocolos de Comando:

cp



Copy (Copiar)

Client

cp [path/file] [destination]

"OK: Files copied"

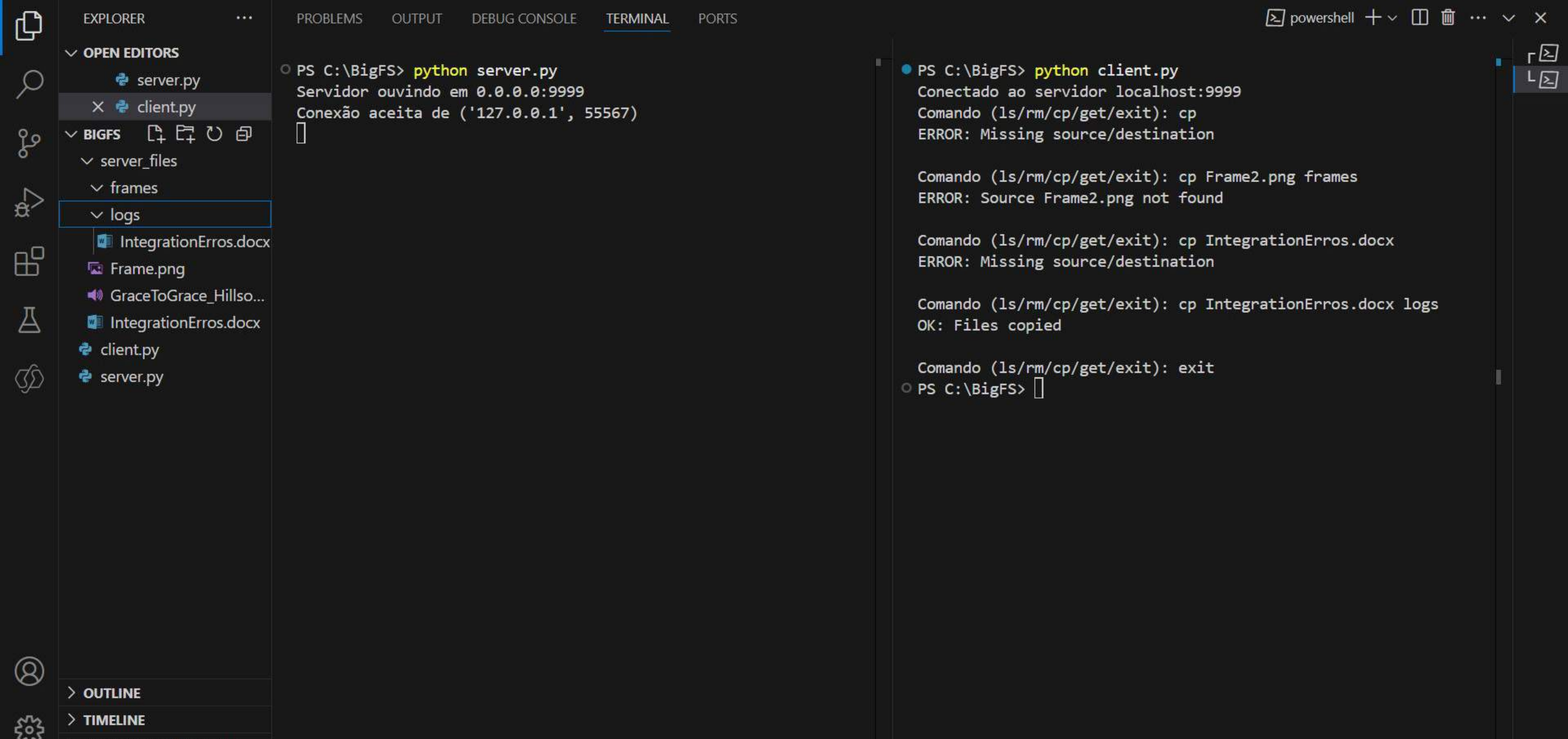
"ERROR: Source [X] not found"

Server

Validação de existência

Cria diretório destino

Copia binariamente

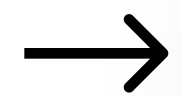


Demonstração rm

BigFS

Protocolos de Comando:

get



Get (Obter)

Client

get [path/file]

[directoryName]

ERROR: Missing path

Transferência concluída

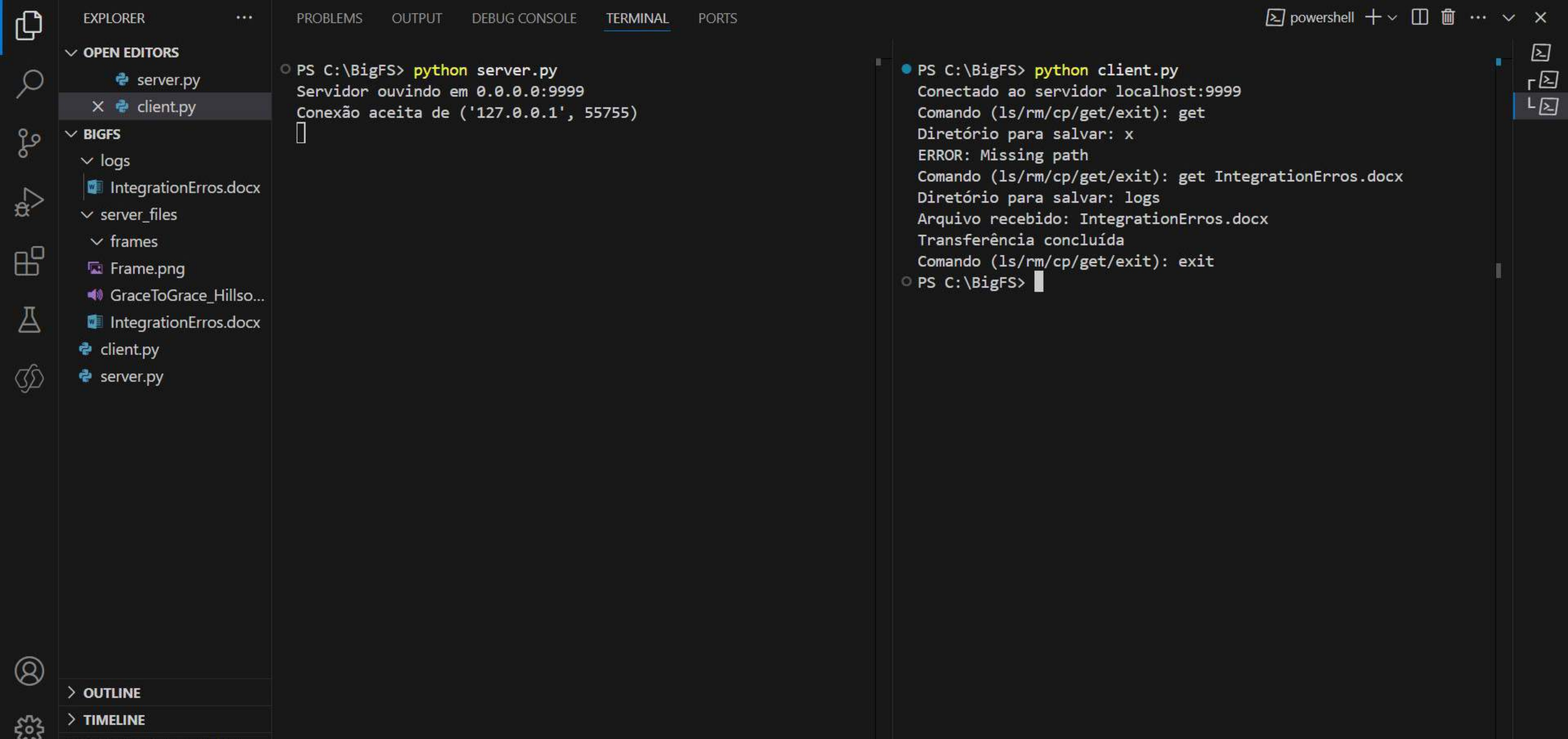


Server

“Diretório para salvar:”

Valida a existência

Retorna: dados binários



Demonstração get

Tratamento de Exceções (Erros)

Comando Inválido

Em caso de comandos não mapeados (inválidos):

ERROR: Invalid command

Caminho Inexistente

Em caso de caminhos inexistentes no diretório local:

ERROR: Path/file not found

Falha na Cópia

Erro ao tentar efetuar uma cópia de um arquivo existente:

ERROR: Copy failed

A utilização de verificações prévias antes de operações e mensagens padronizadas para clientes são estratégias de validação do código.

Segurança & Limitações

No escopo inicial (**etapa 1**) não temos a implementação de aspectos relativos à segurança:

1. Não há autenticação de usuários;
2. Dados trafegam sem criptografia;
3. Não há controle de permissões.

Quanto as limitações atuais dessa etapa:

1. Tamanho máximo de arquivo limitado pela memória;
2. Falta de suporte de operações recursivas.

Segurança

Comando Inválido

Em caso de comandos não mapeados (inválidos):

ERROR: Invalid command

Caminho Inexistente

Em caso de caminhos inexistentes no diretório local:

ERROR: Path/file not found

Falha na Cópia

Erro ao tentar efetuar uma cópia de um arquivo existente:

ERROR: Copy failed

A utilização de verificações prévias antes de operações e mensagens padronizadas para clientes são estratégias de validação do código.

Obrigado!

Acesse: [BigFileS_DFS](#).