

## Relazione Tecnica - Web Server in Python e Sito Statico

### Obiettivo del progetto

L'obiettivo era sviluppare un web server semplice utilizzando Python, in grado di rispondere a richieste HTTP e di servire un sito web statico composto da almeno tre pagine HTML. Il progetto include la gestione delle risposte corrette (200 OK) e degli errori (404 Not Found) in caso di risorse mancanti.

### Struttura del progetto

Il progetto è organizzato in questo modo:

- `server.py`: contiene tutto il codice del server.
- Cartella `www/`: include le tre pagine HTML e i relativi file CSS.
- `server.log`: file che raccoglie un registro delle richieste ricevute.

### Funzionalità implementate

- Il server ascolta sulla porta 8080 in locale (localhost).
- Gestisce correttamente le richieste HTTP di tipo GET.
- Risponde con codice 200 e contenuto corretto se il file richiesto è presente.
- Risponde con codice 404 se il file richiesto non esiste.
- In caso di richieste malformate o di tipo non gestito, restituisce errore 400.
- I tipi MIME sono rilevati automaticamente e usati per impostare l'intestazione della risposta (`Content-Type`).
- Ogni richiesta viene registrata con data, metodo, percorso e codice di risposta.

### Sito web statico

Il sito è costituito da tre pagine collegate tra loro:

- `index.html` (home page)
- `about.html` (chi siamo)
- `contact.html` (pagina contatti)

Tutte le pagine condividono uno stile coerente grazie al file CSS incluso. Il layout è progettato per adattarsi anche a dispositivi con schermi più piccoli, mantenendo leggibilità e struttura.

### **Conclusione**

Il progetto soddisfa i requisiti minimi richiesti e integra funzionalità aggiuntive come il rilevamento automatico dei tipi MIME, la scrittura di log e una struttura HTML adatta anche a schermi mobili. Il server risponde correttamente alle richieste previste e gestisce i principali casi di errore.