

Reliable UDP

Trasferimento file affidabile su protocollo UDP

Luca Santopadre 0257118 | Ingegneria Internet e Web 2019/2020

Sommario

[Premessa 2](#_Toc50996880)

[Architettura 3](#_Toc50996881)

[Server 3](#_Toc50996882)

[Client 3](#_Toc50996883)

[Scelte Progettuali 4](#_Toc50996884)

[Gestione della concorrenza 4](#_Toc50996885)

[Implementazione 5](#_Toc50996886)

[Orientamento alla connessione 5](#_Toc50996887)

[Trasferimento dati affidabile 5](#_Toc50996888)

[Comandi 5](#_Toc50996889)

[List 5](#_Toc50996890)

[Get 5](#_Toc50996891)

[Put 5](#_Toc50996892)

[Manuale di utilizzo, configurazione 6](#_Toc50996893)

[Esempi 7](#_Toc50996894)

[Analisi delle prestazioni 8](#_Toc50996895)

# Premessa

Il presente documento intende descrivere ed analizzare il progetto finale del corso di Ingegneria di Internet e web 2019/2020.

Lo scopo de progetto è quello di progettare ed implementare in linguaggio C usando l’API del socket di Berkeley un’applicazione client-server per il trasferimento di file che impieghi il servizio di rete senza connessione (socket tipo SOCK\_DGRAM, ovvero UDP come protocollo di strato di trasporto). Il software deve permettere:

• Connessione client-server senza autenticazione;

• La visualizzazione sul client dei file disponibili sul server (comando list);

• Il download di un file dal server (comando get);

• L’upload di un file sul server (comando put);

• Il trasferimento file in modo affidabile.

# Architettura

## Server

Per l’esercizio di questo software si utilizza un’architettura client-server

## Client

# Scelte Progettuali

## Gestione della concorrenza

## Limitazioni riscontrate

# Implementazione

## Orientamento alla connessione

## Trasferimento dati affidabile

## Comandi

### List

### Get

### Put

# Manuale di utilizzo, configurazione

# Esempi

# Analisi delle prestazioni