

**Download manažér**

SEMESTRÁLNA PRÁCA

Vypracovali: **Lukáš Strmý, Filip Cirbus**

Študijná skupina: **5ZYI35**

Predmet: **Princípy operačných systémov**

Cvičiaci: **Ing. Ján Rabčan, PhD.**

Obsah

[1. Programátorská dokumentácia 3](#_Toc155437235)

[1.1. Štruktúra projektu 3](#_Toc155437236)

[1.2. Použitie socketov 4](#_Toc155437237)

[1.3. Použitie vlákien 5](#_Toc155437238)

[1.4. Synchronizačný problém 6](#_Toc155437239)

[2. Používateľská dokumentácia 7](#_Toc155437240)

# Programátorská dokumentácia

## Štruktúra projektu

Obrázok, na ktorom je text, snímka obrazovky, písmo, číslo

Automaticky generovaný popis

Používame len jednu triedu a tou je client.

V klientovi sa nachádza viacero funkcií. Každá plní iný účel.

Hlavná funkcia main v ktorej sa nachádza cyklus, ktorý používateľovi stále dookola ponúka možnosti aplikácie, až kým sa používateľ nerozhodne aplikáciu ukončiť.

Funkcia logger zapisuje do súboru logger.txt informácie o stiahnutých súboroch a pri inom nastavení tieto informácie vypíše na konzolu.

Funkcia parse\_url rozčleňuje url na menšie celky

Funkcia directoryExists slúži na overenie, či daný adresár existuje.

Funkcia downFolderPath umožňuje čítať a zapisovať do súboru, kde sa nachádza uložená cesta k priečinku do ktorého je nastavené sťahovanie.

Funkcia extract\_filename slúži na extrahovanie názvu súboru zo zadanej url adresy.

Funkcia create\_ssl\_context vytvára a inicializuje SSL/TLS kontext pre klientskú stranu.

Funkcia parseFtpUrl rozloží url adresu pri FTP spojení.

Funkcia threadFunc je funkcia vlákna, v ktorej prebieha sťahovanie súboru.

## Použitie socketov

 **Vytvorenie socketu a pripojenie k serveru**:

* Socket sa vytvára pomocou socket(AF\_INET, SOCK\_STREAM, 0).
* Nastavuje sa adresa serveru cez štruktúru struct sockaddr\_in.
* Pripája sa k serveru pomocou connect(sockfd, (struct sockaddr \*) &server\_addr, sizeof(server\_addr)).

 **Posielanie HTTP GET požiadavky**:

* HTTP GET požiadavka je odosielaná buď pomocou dprintf pre HTTP alebo pomocou SSL\_write pre HTTPS.

 **Prijímanie dát**:

* Dáta sú prijímané cez read pre HTTP alebo SSL\_read pre HTTPS.

 **Uzatvorenie socketu a SSL spojenia**:

* Socket je uzavretý pomocou close(sockfd).
* Pre HTTPS je potrebné tiež uzavrieť SSL spojenie pomocou SSL\_free a SSL\_CTX\_free.

## Použitie vlákien

Aby bola aplikácia použiteľná aj po tom, čo používateľ dá sťahovať súbor, tak sú v nej implementované vlákna.

* **Štruktúra VlaknoInfo**:
  + Každé vlákno má svoju štruktúru VlaknoInfo, ktorá obsahuje identifikátor vlákna (pthread\_t id), časovač (int timer) a URL pre sťahovanie (const char \*url).
* **Pole VlaknoInfo vlakna[MAX\_VLAKIEN]**:
  + Pole štruktúr VlaknoInfo slúži na uchovávanie informácií o vytvorených vláknach. Jeho veľkosť je obmedzená na MAX\_VLAKIEN, čo znamená maximálny počet sťahovaní, ktoré môžu byť spustené súčasne.
* **Hlavná funkcia main**:
  + V nekonečnom cykle (s výberom možností od 1 do 5) používateľ zadáva rôzne príkazy.
  + Ak používateľ vyberie možnosť sťahovania súboru alebo plánovania sťahovania, program vytvorí nové vlákno pre každý sťahovaný súbor.
    - Informácie o vlákne (časovač, URL atď.) sa ukladajú do štruktúr VlaknoInfo v poli vlakna.
    - Vytvorenie nového vlákna sa vykonáva pomocou pthread\_create, kde sa ako vstup pre funkciu vlákna vlaknoFunkcia odovzdá štruktúra VlaknoInfo.
    - Po vytvorení nového vlákna je inkrementovaný počet aktívnych vlákien pocetVlakien.
* **Správa maximálneho počtu sťahovaní**:
  + Ak sa dosiahne maximálny počet sťahovaní (MAX\_VLAKIEN), program informuje používateľa, že nemôže pridať ďalšie vlákno, kým sa nejaké z aktívnych vlákien neukončí.

## Synchronizačný problém

V aplikácii sme zabránili možnosti že dve vlákna sa snažia zapisovať informácie do toho istého súbor naraz. Na to sme použili príkaz pthread\_mutex\_lock(info->mutex) ktorým zamedzíme ostatným vláknam prístup do sťahovaného súboru, zatiaľ čo im naďalej umožňujeme sťahovať ich vlastné súbory. Po dokončení sťahovania nesmieme zabudnúť mutex odomknúť pomocou pthread\_mutex\_unlock(info->mutex).

# Používateľská dokumentácia

**POS Download Manager** je jednoduchý nástroj na sťahovanie súborov z internetu pomocou protokolov HTTP, HTTPS, FTP. Tento manažér vám umožňuje sťahovať súbory okamžite alebo ich plánovať na neskorší čas. Navyše môžete spravovať zložku, do ktorej sa súbory sťahujú, a prezrieť si históriu svojich sťahovaní.

### **Použitie**

### Sťahovanie súborov

Ak chcete okamžite stiahnuť súbor, vyberte možnosť 1 a zadajte URL súboru. Program vytvorí nové vlákno pre každý sťahovaný súbor.

### Plánovanie sťahovania

Vyberte možnosť 2, ak chcete plánovať sťahovanie na neskorší čas. Zadajte časový interval (v minútach) a potom URL súboru. Program automaticky spustí sťahovanie po uplynutí zadaného intervalu.

### Správa zložky pre sťahovanie

Ak chcete zmeniť zložku pre sťahovanie, vyberte možnosť 3. Program vás vyzve, aby ste zadali novú cestu k zložke.

### Zobrazenie histórie sťahovaní

Vyberte možnosť 4, ak chcete zobraziť históriu sťahovaní. Program zobrazí všetky dokončené sťahovania vrátane času a názvu súboru.

### Ukončenie programu

Pre ukončenie programu vyberte možnosť 5. Program sa ukončí a počká na dokončenie všetkých aktuálnych sťahovaní.

Dúfame, že **POS Download Manager** uľahčí vaše sťahovacie operácie!