

UNIVERSITA

PROGETTO A CURA DI :
LUISI FRANCESCOMARIA 0124002603
MASCOLO ELIODORO 0124002547

Introduzione

Il sistema di gestione degli esami universitari è progettato per facilitare la comunicazione tra la segreteria universitaria, gli studenti e il server universitario. Le funzionalità principali includono l'inserimento e la gestione degli esami, la prenotazione degli esami da parte degli studenti e la gestione delle richieste da parte del server universitario.

REQUISITI

Segreteria

- 1) Inserisce gli esami nel sistema.
- 2) inoltra le richieste di prenotazione degli studenti.
- 3) Fornisce le date degli esami disponibili agli studenti.

Studente

- 1) Richiede alla segreteria informazioni sugli esami disponibili.
- 2) Invia richieste di prenotazione alla segreteria.

Server Universitario

- 1) Gestisce l'aggiunta di nuovi esami.
- 2) Gestisce le prenotazioni degli esami.

GESTIONE DELLA SEGRETERIA

Segreteria.c

Il file Segreteria.c gestisce le interazioni principali della segreteria con il server universitario e gli studenti. La prima operazione è l'allocazione della memoria per una struttura (Thread_Segreteria) che contiene gli argomenti da passare ai vari thread. Successivamente, viene creato un socket per la segreteria, configurato con l'indirizzo specifico e con opzioni per permettere il riutilizzo dell'indirizzo stesso.

Questo socket viene quindi associato all'indirizzo configurato.

Il programma prosegue creando e configurando un altro socket per connettersi al server universitario. Viene poi inizializzata una serie di variabili per gestire le connessioni degli studenti. Per gestire l'input della segreteria, viene avviato un thread dedicato. Il socket della segreteria viene messo in ascolto per accettare le connessioni degli studenti. Ogni volta che uno studente si connette, viene avviato un thread separato per gestire la connessione specifica dello studente.

Segreteria_Helper.c

Thread Gestione Studenti

Riceve i dati dagli studenti, li inoltra al server e riceve la risposta.

Gestisce la comunicazione tra gli studenti e il server.

Gestisce la chiusura delle connessioni in caso di disconnessione.

Thread Gestione Input Segreteria

Permette alla segreteria di selezionare operazioni come aggiunta di esami o appelli.

Legge l'opzione selezionata e esegue la richiesta corrispondente.

Assicura la pulizia del buffer di input per evitare errori.

Funzioni di Selezione Operazioni Segreteria

Presentano le opzioni disponibili per la segreteria.

Leggono e interpretano l'opzione selezionata dall'utente.

Funzioni di Aggiunta di Esami e Appelli

Leggono i dettagli necessari per l'aggiunta di esami o appelli dalla segreteria.

Formattano i dati e li inviano al server universitario per l'elaborazione.

Ricevono una conferma dal server e comunicano il risultato alla segreteria.

GESTIONE DEL SERVER

Il codice del Server Principale (Server.c) gestisce la comunicazione tra la segreteria e il sistema di gestione degli esami universitari. Utilizza socket per l'ascolto delle connessioni, si connette al database SQLite per gestire dati degli studenti, esami e prenotazioni, e supporta operazioni come verifica credenziali, visualizzazione appelli, e aggiunta prenotazioni.

Il codice delle Funzioni di Supporto (Server_Helper.c) implementa operazioni specializzate come la verifica credenziali, visualizzazione appelli, e gestione prenotazioni. Le funzioni integrano dettagliati controlli di errore per garantire un funzionamento affidabile del sistema di gestione esami universitari.

Studente_Helper.c fornisce funzioni per facilitare le operazioni degli studenti in un sistema accademico. Ecco un riassunto delle funzioni principali:

Login:

Richiesta_Visualizzazione_Appelli:

Richiesta_Prenotazione_Appello:

Costruisce e invia la richiesta di prenotazione al server.

Riceve la risposta dal server e gestisce diversi scenari:

successo della prenotazione, appello già prenotato, corso non valido, appello non valido.

GESTIONE STUDENTE

Il file `Studente.c` contiene la funzione principale che gestisce il ciclo di vita del programma studente.

Main:

Avvia il ciclo principale del programma.

Tenta di effettuare il login dello studente utilizzando la funzione `Login`.

Se il login ha successo, mostra un menu per selezionare le operazioni:

Visualizzare appelli disponibili.

Prenotare un appello.

Uscire dal programma.

Gestisce le opzioni selezionate dall'utente, chiamando le funzioni appropriate

(`Richiesta_Visualizzazione_Appelli`, `Richiesta_Prenotazione_Appello`).

Se l'utente sceglie di uscire, chiude il socket, libera la memoria e termina il programma.

Se il login fallisce, permette di ritentare.

Queste funzioni collaborano per fornire un'interfaccia semplice e intuitiva agli studenti per interagire con il sistema accademico.