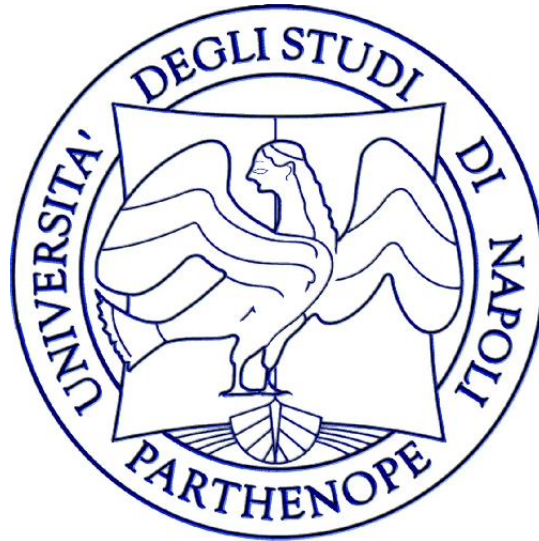


UNIVERSITÀ DEGLI STUDI DI NAPOLI “PARTHENOPE”  
FACOLTÀ DI SCIENZE E TECNOLOGIE  
CORSO DI LAUREA IN INFORMATICA



PROGETTO DI RETI DI CALCOLATORI

Università

DOCENTE

Emanuel Di Nardo

STUDENTI

Roberto Della Corte - 0124002666

Giuseppe Cristiano - 0124002534

Raffaele Lettieri - 0124002631

Anno Accademico 2023-2024

# Indice

<b>1</b>	<b>Traccia Progetto</b>	<b>1</b>
<b>2</b>	<b>Entità del Progetto</b>	<b>3</b>
2.1	Architettura e Diagramma . . . . .	4
<b>3</b>	<b>Manuale utente</b>	<b>5</b>
3.1	Server . . . . .	5
3.2	Segreteria . . . . .	5
3.3	Studente . . . . .	6
<b>4</b>	<b>Database MYSQL</b>	<b>7</b>
4.1	Diagramma del Database . . . . .	7
4.2	studente . . . . .	8
4.3	prenotazione . . . . .	8
4.4	appelli . . . . .	8
4.5	esami . . . . .	9

# Capitolo 1

## Traccia Progetto

Scrivere un'applicazione client/server parallelo per gestire gli esami universitari.

- **Segreteria:**

- Inserisce gli esami sul server dell'università (salvare in un file o conservare in memoria il dato)
- Inoltra la richiesta di prenotazione degli studenti al server universitario
- Fornisce allo studente le date degli esami disponibili per l'esame scelto dallo studente

- **Studente:**

- Chiede alla segreteria se ci siano esami disponibili per un corso
- Invia una richiesta di prenotazione di un esame alla segreteria

- **Server universitario:**

- Riceve l'aggiunta di nuovi esami
- Riceve la prenotazione di un esame

Il server universitario ad ogni richiesta di prenotazione invia alla segreteria il numero di prenotazione progressivo assegnato allo studente e la segreteria a sua volta lo inoltra allo studente

Se la segreteria non risponde alla richiesta dello studente questo deve ritentare la connessione per 3 volte. Se le richieste continuano a fallire allora aspetta un tempo random e ritenta. Simulare un timeout della segreteria in modo da arrivare a testare l'attesa random.

# Capitolo 2

## Entità del Progetto

Le entità, parti che interagiscono tra di loro, all'interno del nostro progetto sono:

- **Server:** il server, funge da parte server, in un'architettura client/server, per la segreteria inviando risposte alla componente segreteria. Il server ha il compito di aggiungere un appello di un esame quando richiesto dalla segreteria, ed inoltre permette la prenotazione di studenti all'appello d'esame, quando uno studente richiede una prenotazione alla segreteria;
- **Segreteria:** la segreteria ha il compito sia da client che da server. E' client per quanto riguarda la sua interazione con il server, mentre è server per quanto riguarda la sua interazione con lo studente. Ha diversi compiti tra cui: richiedere al server l'aggiunta di un'appello di un'esame, permettere allo studente di potersi prenotare ad un appello inoltrando la richiesta al server universitario e far visualizzare agli studenti gli appelli disponibili per il loro corso di studi o per un determinato esame;
- **Studente:** lo studente è una componente che funge da client per la segreteria ed ha la possibilità di: richiedere alla segreteria la visualizzazione degli appelli per quanto riguarda il suo corso di studi o di un esame specifico e la possibilità di potersi prenotare ad un appello inoltrando la richiesta alla segreteria.

## 2.1 Architettura e Diagramma

E' possibile disegnare un diagramma che rappresenti l'architettura del nostro progetto



Le frecce continue indicano il collegamento tra le diverse entità, mentre quelle tratteggiate sono come le diverse entità comunicano tra di loro.

# Capitolo 3

## Manuale utente

### 3.1 Server

Per poter runnare il server, apriamo il terminale. Entrando nella cartella in cui c'è il file `server.c`, eseguiamo la compilazione di quest'ultimo tramite il

comando:

```
gcc server.c -o server -lmysqlclient
```

Questa istruzione genera un file `server` che sarà eseguito tramite il comando:

```
./server
```

### 3.2 Segreteria

Per poter runnare la segreteria, apriamo un altro terminale. Entrando sempre nella cartella in cui c'è il file `segreteria.c`, eseguiamo la compilazione di quest'ultimo tramite il comando:

```
gcc segreteria.c -o segreteria -lmysqlclient
```

Questa istruzione genera un file `segreteria` che sarà eseguito tramite il comando:

```
./segreteria "Indirizzo IP"
```

Per poter eseguire la segreteria, bisogna passare l'indirizzo IP, in questo caso l'indirizzo localhost `127.0.0.1`

### 3.3 Studente

Per poter runnare lo studente, aprimao un terzo terminale. Entrando sempre nella cartella in cui c'è il file `studente.c`, esqguiamo la comppilazione di ques'ultimo tramite il comando:

```
gcc studente.c -o studente
```

Questa istruzione genera un file `studente` che sarà eseguito tramite il comando:

```
./studente "Indirizzo IP"
```

Per poter eseguire la studente, anche esso ha bisogno di un indirizzo IP, in questo caso l'indirizzo localhost `127.0.0.1`

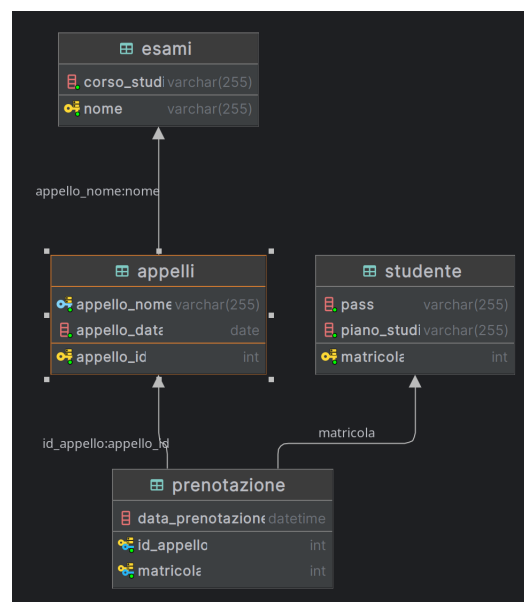


# Capitolo 4

## Database MYSQL

Il database all'interno del progetto viene utilizzato per lo storing dei dati necessari, e viene utilizzato il DBMS MYSQL in localhost.

### 4.1 Diagramma del Database



Adesso andremo a descrivere i vari campi, PK e FK del nostro database.

## 4.2 studente

La tabella **studente** contiene i campi:

- **matricola**: campo che rappresenta la matricola dello studente: PK;
- **piano studi**: campo che rappresenta il piano di studi dello studente;
- **pass**: campo che rappresenta la password dello studente.

## 4.3 prenotazione

La tabella **prenotazione** contiene i campi:

- **id appello**: id dell'appello a cui lo studente si prenota: FK di appelli riferendosi ad appelloid e PK;
- **matricola**: matricola dello studente che ha effettuato la prenotazione: FK di studente riferendosi a matricola e PK anch'essa;
- **data prenotazione**: data in cui lo studente ha effettuato la prenotazione.

## 4.4 appelli

La tabella **appelli** contiene i seguenti campi:

- **appello id**: id dell'appello: PK;
- **appello nome**: nome dell'esame di cui si fa l'appelloFK di esami riferendosi a nome;
- **appello data**: data in cui si svolge l'appello.

## 4.5 esami

La tabella **esami** contiene i campi:

- **nome**: nome dell'esame: PK;
- **corso studi**: corso di studi al quale appartiene e aiuta agli studenti a prenotarsi solo agli esami del loro corso di studi.