

TÀI LIỆU THIẾT KẾ PHẦN MỀM SCORE SCHOOL SYSTEM

Tác giả: Nguyễn Đình Nam
nguyendinhnam.working@gmail.com
26/09/2019

MỤC LỤC

DANH MỤC HÌNH VẼ
DANH MỤC BẢNG BIỂU

CHƯƠNG 1: GIỚI THIỆU

- 1.1 Mục đích
- 1.2 Phạm vi
- 1.3 Thuật ngữ

CHƯƠNG 2: TỔNG QUAN HỆ THỐNG

- 2.1 Thông tin chung
- 2.2 Các yêu cầu hệ thống (SRS)
- 2.3 Các yêu cầu phần mềm (SRS)
- 2.4. Tổng quan kiến trúc phần mềm (SAD)

CHƯƠNG 3: THIẾT KẾ MODULE PHẦN MỀM (SDD)

- 3.1 Thiết kế phía Client
- 3.2 Thiết kế phía Server

CHƯƠNG 4: THIẾT KẾ DỮ LIỆU

- 4.1 Kiểu dữ liệu
- 4.2 Dữ liệu tĩnh
- 4.3 Dữ liệu động

KẾT LUẬN

TÀI LIỆU THAM KHẢO

DANH MỤC HÌNH VẼ

Hình 2.1: Mô hình TCP Client-Server

Hình 2.2: Kiến trúc gọi - trả lại

Hình 2.3: Kiến trúc phần mềm quản lý điểm trường học “Score School Software”

DANH MỤC BẢNG BIỂU

Bảng 1.1: Các thuật ngữ viết tắt

Bảng 2.1: Đặc điểm của kiểu kiến trúc gọi - trả lại

CHƯƠNG 1: GIỚI THIỆU

1.1 Mục đích

Với sự phát triển nhanh chóng của công nghệ thông tin và truyền thông (ICT), nhiều lĩnh vực trong đời sống đã được tin học hóa, giúp cho quá trình xử lý công việc nhanh và đơn giản hơn, giúp cho tiết kiệm rất nhiều thời gian và tiền bạc.

Tra cứu điểm số là nhu cầu thiết yếu của học sinh, sinh viên trong các cơ sở giáo dục (trường học, học viện, trung tâm,...). Hiện nay, thông tin về người học đã được số hóa hoàn toàn và được lưu trữ tại các trung tâm dữ liệu - Data Center, đóng vai trò là Server. Người học từ bất cứ đâu đều có thể truy xuất thông tin về tài khoản, điểm số của mình thông qua các thiết bị đầu cuối (máy tính, điện thoại) thông qua mạng Internet với một tài khoản duy nhất.

Với nhu cầu như vậy, chúng tôi đề xuất một phần mềm quản lý thông tin người học (Tên, mã sinh viên, điểm số,...), là “Score School System” nhằm mục tiêu hướng đến sự thuận tiện và đơn giản trong quản lý của các cơ sở giáo dục.

1.2 Phạm vi

Tài liệu nhằm mô tả thiết kế của phần mềm tra cứu điểm thi. Ứng dụng sẽ được triển khai trong các cơ sở giáo dục các cấp. Đối tượng sử dụng là các kỹ thuật viên quản trị hệ thống cơ sở dữ liệu của trường học và các bạn học sinh, sinh viên có nhu cầu truy xuất thông tin của mình.

1.3 Thuật ngữ

Bảng 1.1: Các thuật ngữ viết tắt.

STT	Từ viết tắt	Tiếng anh	Nghĩa tiếng việt
1	SRS	System Requirement Specification	Đặc tả yêu cầu hệ thống
2	SRS	Software Requirement Specificaion	Đặc tả yêu cầu phần mềm
3	SAD	Software Architecture Design	Thiết kế kiến trúc phần mềm
4	SDD	Software Detailed Design	Thiết kế chi tiết phần mềm
5	ICT	Information and Communication Technology	Công nghệ thông tin và truyền thông
6	TCP	Transmission Control Protocol	Giao thức điều khiển truyền dẫn

CHƯƠNG 2: TỔNG QUAN HỆ THỐNG

2.1 Thông tin chung

2.1.1 Bài toán đặt ra: [1]

Thiết kế phần mềm quản lý điểm số cho trường học. Dữ liệu được lưu tại trung tâm - Data Center (Server). Người dùng (Client) sẽ truy xuất thông tin thông qua một tài khoản duy nhất. Mô hình sử dụng là TCP Server – Client có 3 tiến trình như sau:

TCP user server: đóng vai trò là TCP server, chứa thông tin của các Client: client_id, client_password, client_score,... Server sẽ trả về kết quả truy nhập và điểm số khi có yêu cầu truy nhập từ Client.s

TCP user client 1: đóng vai trò là TCP user client, gửi yêu cầu truy nhập và yêu cầu điểm số của Client 1 đến Server.

TCP user client 2: đóng vai trò là TCP user client, gửi yêu cầu truy nhập và yêu cầu điểm số của Client 2 đến Server.

2.1.2 Luồng hoạt động [1]

TCP user client 1 (C1), TCP user server (S)

1. C1 gửi cho S bản tin LOG_IN_REQUEST (S xử lý trên Thread 1)
2. S gửi cho C1 bản tin LOG_IN_SUCCESS
3. C1 gửi cho S bản tin SCORE_REQUEST
4. S gửi cho C bản tin SCORE_SUCCESS
5. C1 gửi cho S bản tin LOG_OUT_REQUEST
6. S gửi cho C1 bản tin LOG_OUT_SUCCESS
7. C1 gửi cho S bản tin SCORE_REQUEST
8. S gửi cho C1 bản tin SCORE_FAILURE

TCP user client 2 (C2), TCP user server (S)

1. C2 gửi cho S bản tin LOG_IN_REQUEST (S xử lý trên Thread 2)
2. S gửi cho C2 bản tin LOG_IN_SUCCESS

2.2 Các yêu cầu hệ thống (SRS - System Requirement Specification)

- Hệ điều hành Linux (Ubuntu):

Đề xuất: Ubuntu 18.04.2. LTS, core i5, ram 8G.

2.3 Các yêu cầu phần mềm (SRS - Software Requirement Specification)

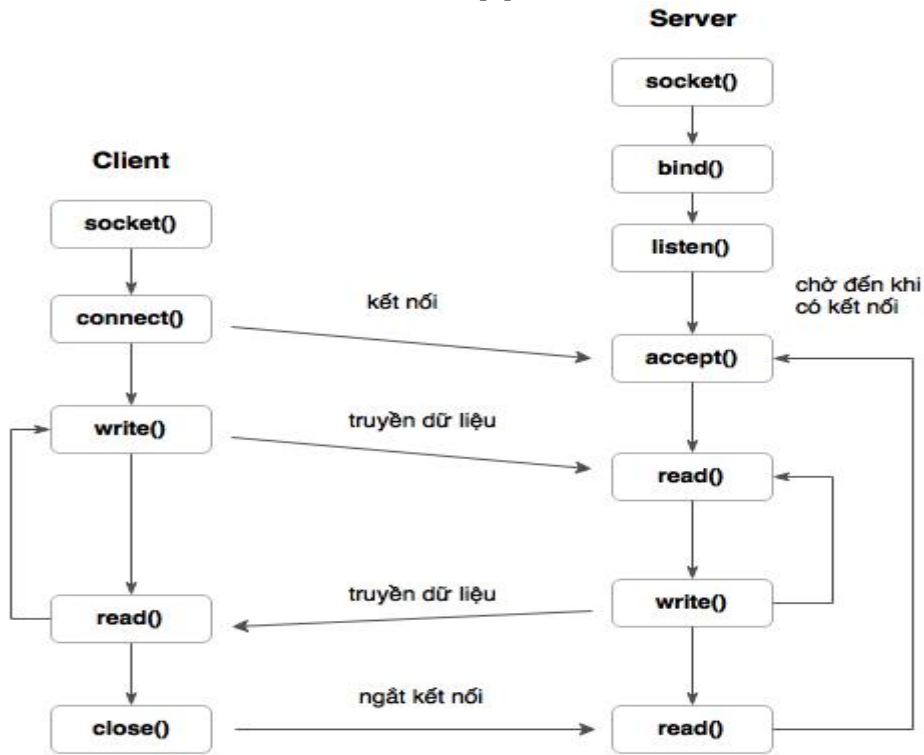
- Yêu cầu về tính bảo mật, Client cần có tài khoản để đăng nhập vào hệ thống. Chức năng Login (student_id , password)

- Khả năng truy xuất thông tin trả về dữ liệu người học (Họ tên, Mã sinh viên, Điểm thi các môn,...)

- Ngôn ngữ lập trình C
- Giao diện người sử dụng (Client) thông qua dòng lệnh (Command Line)
- Trong mô hình TCP Server - Client đơn giản thì Server chỉ có thể xử lý một Client tại một thời điểm. Tuy nhiên, yêu cầu phần mềm này là xử lý đa người dùng với một Server (mỗi Client trên một Thread).

2.4. Tổng quan kiến trúc phần mềm (SAD - Software Architecture Design)

2.4.1 Mô hình TCP Server – Client [3]



Hình 2.1: Mô hình TCP Server - Client

- Đối với **Client**:

B1: Tạo socket để kết nối qua hàm socket().

B2: Kết nối tới server nhờ hàm connect(). Hàm này thông thường sẽ block cho tới khi nào kết nối được.

B3: Gửi và nhận thông tin qua hàm send/recv.

- Đối với **Server**:

B1: Tạo socket để kết nối qua hàm socket()

B2: Gắn socket vừa tạo với địa chỉ của server. Dùng hàm bind().

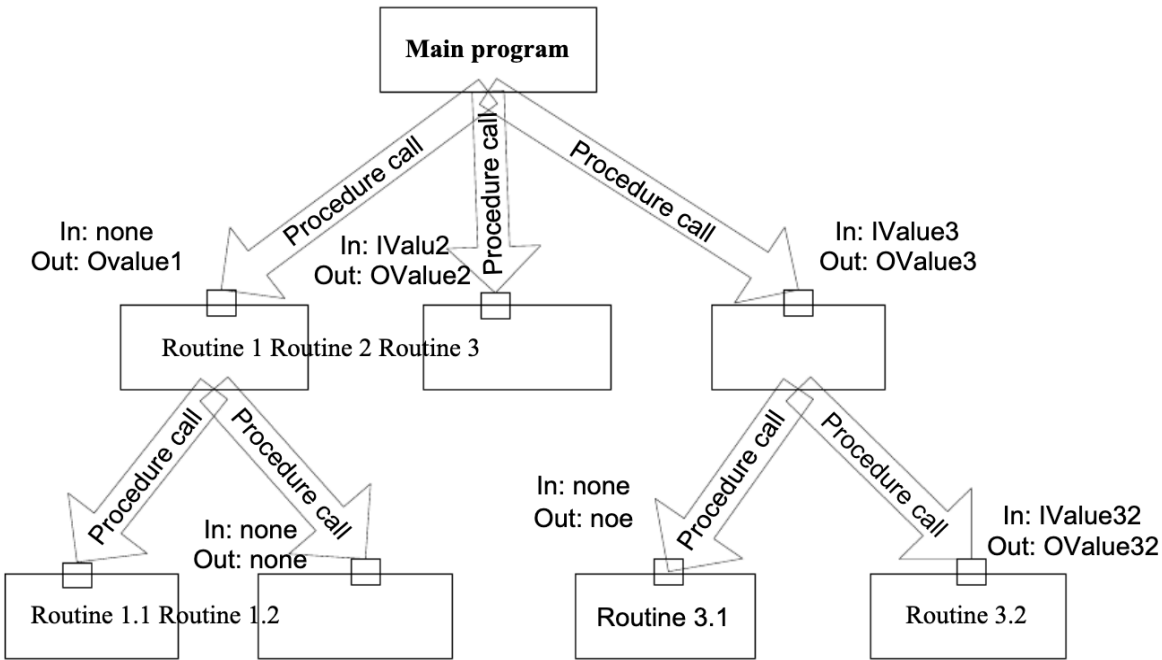
B3: Thiết lập socket để chờ kết nối. Dùng hàm listen().

B4: Chấp nhận kết nối từ client. Hàm accept() thông thường sẽ bị block tới khi nào có kết nối tới.

B5: Gửi và nhận thông tin qua hàm send()/recv().

2.4.2 Kiến trúc gọi - Trả lại (Main Program and subroutines) [4]

Phần mềm tra cứu điểm trường học được thiết kế dựa trên kiến trúc gọi - trả lại (Main Program and Subroutines) được minh họa trên hình 2.2.
Đặc điểm của kiểu kiến trúc gọi - trả lại được mô tả chi tiết trong bảng 2.1.

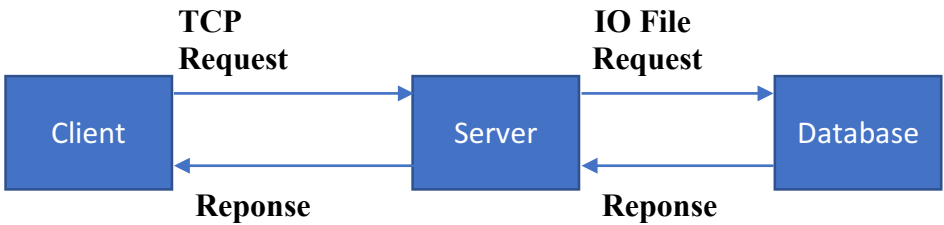


Hình 2.2: Kiến trúc gọi - trả lại

Kiểu kiến trúc	Program and subroutines
Đặc điểm	Theo phong cách hướng thủ tục, chức năng lớn được phân rã thành các chức năng con. Chương trình gồm một chương trình chính và gọi tới các chương trình con.
Thành phần	Chương trình chính và các chương trình con.
Kết nối	Lời gọi hàm.
Lý do lựa chọn	Đơn giản, dễ hiểu. Dễ sử dụng với các hệ thống vừa và nhỏ, độ phức tạp không lớn.

Bảng 2.1: Đặc điểm của kiểu kiến trúc gọi - trả lại

2.4.3 Kiến trúc phần mềm quản lí điểm trường học - “Score School Software”



Hình 2.3: Kiến trúc phần mềm quản lí điểm trường học - “Score School Software”

CHƯƠNG 3: THIẾT KẾ MODULE PHẦN MỀM (SDD)

3.1 Thiết kế phía Client

- Chức năng:

- + Thông qua giao diện dòng lệnh để giao tiếp với Server.
- + Gửi yêu cầu đăng nhập: Log In Request
- + Gửi thông tin đăng nhập: Username và Password
- + Gửi yêu cầu truy xuất điểm số: Score Request
- + Nhận bản tin từ Server để xử lý và hiện thị ra giao diện console (Xử lý tách, ghép xâu).
- + Hai Client cùng giao tiếp đồng thời với một Server.

- Bản tin:

Bản tin được gửi tới Server qua giao thức TCP.

Quy ước: 0 -> Đăng nhập; 1 -> Truy xuất thông tin và điểm số; 2 -> Thoát

- + Bản tin đăng nhập: **0[username]:[password]**

Ví dụ: 0DinhNamHMU:mainam231299

- + Bản tin truy xuất thông tin: **1[student_ID]:[subject]**

Ví dụ: 1B15DCVT262:Physic

Với, username là tên người dùng

password là mật khẩu

student_ID là mã sinh viên

subject là môn học

3.2 Thiết kế phía Server

- Chức năng:

- + Nhận bản tin từ Client thông qua giao thức TCP. Xử lý bản tin (Xử lý tách, ghép xâu) để xác định yêu cầu, xử lý yêu cầu và truyền bản tin hồi đáp lại Client.
- + Truy xuất thông tin từ Database bằng giải thuật tìm kiếm.

- Bản tin:

- + Bản tin yêu cầu đăng nhập để tiếp tục

Ví dụ: "Hi, I'm a Server. Please log in to continue! [0Username:Password]"

- + Bản tin thông báo đăng nhập thành công

Ví dụ: "Congratulations. Successful login. Can I help you, please! Nguyen Dinh Nam, B15DCVT262"

- + Bản tin thông báo lỗi tên tài khoản hoặc mật khẩu

Ví dụ: "Username or password incorrect. Please log in to continue! [0Username:Password]"

- + Bản tin trả về thông tin và điểm số

Ví dụ: "Your full name: Nguyen Dinh Nam, Your student ID: B15DCVT262, Physic: 8 score."

- + Bản tin thông báo thoát thành công

Ví dụ: "You haved logged out. Please log in to continue! [0Username:Password]"

- Database:

Cơ sở dữ liệu được lưu trong file .txt, Server truy xuất bằng hàm vào ra IO File.

CHƯƠNG 4: THIẾT KẾ DỮ LIỆU

4.1 Dữ liệu tĩnh

Các kiểu dữ liệu cơ bản trong ngôn ngữ lập trình C (int, float, double, char,...)

Cấu trúc dữ liệu kiểu Structure để lưu trữ thông tin sinh viên.

Ví dụ:

```
typedef struct Student {  
    char username[MAX];  
    char password[MAX];  
    char fullname[MAX];  
    char studenID[MAX];  
    float math;  
    float physic;  
    float chemistry;  
} student;
```

4.2 Dữ liệu động

Kiểu dữ liệu con trỏ. Khởi tạo mảng động

Ví dụ:

```
char *greeting_Message = "Hi, I'm a Server. Please log in to continue! [0Username:Password]\n";
```

KẾT LUẬN

Trên đây là tài liệu thiết kế kiến trúc (SAD - Software Architecture Design) của phần mềm tra cứu điểm số trường học - “Score School System”. Tài liệu đã đạt được các yêu cầu của một thiết kế kiến trúc cơ bản, thực hiện chuyên hóa các đặc tả yêu cầu phần mềm (SRS - Software Requirement Specification) thành các mô tả thiết kế kiến trúc. Đồng thời, đưa ra bản thiết kế chi tiết cho phần mềm, phục vụ cho quá trình cài đặt chương trình được dễ dàng thuận tiện.

THAM KHẢO

[1] “Research & Development the Protocol in 4G-LTE”, tuandn3@viettel.com.vn

[2] “Socket Programming in C/C++: Handling multiple clients on server without multi threading”, geeksforgeeks.org

[3] “TCP Server-Client implementation in C”, geeksforgeeks.org

[4] “Nghiên cứu và thiết kế kiến trúc phần mềm cho các hệ thống lớn và phức tạp”, Vũ Văn Lĩnh, Đại học Công nghệ, 2011