**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

**2.1/ Ứng Dụng Streaming Nhạc Trực Tuyến**

Ứng dụng streaming nhạc trực tuyến là ứng dụng phần mềm cho phép người dùng truy cập và phát nhạc từ một kho dữ liệu âm nhạc trực tuyến. Thay vì tải xuống hoặc lưu trữ âm nhạc trên thiết bị của mình, người dùng có thể nghe nhạc trực tuyến thông qua kết nối internet. Ứng dụng này thường cung cấp một giao diện dễ sử dụng cho phép người dùng tìm kiếm, chọn lựa và phát nhạc từ các bài hát, album, hoặc danh sách phát khác nhau . [1]

**2.2/ Nền tảng .NET:**

- .NET là một nền tảng ứng dụng miễn phí và mã nguồn mở được hỗ trợ bởi Microsoft. Điều này làm cho nó trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng streaming nhạc trực tuyến. [2]

- Ngôn ngữ chính được sử dụng trong .NET là C#, một ngôn ngữ hiện đại và mạnh mẽ. C# cung cấp các tính năng linh hoạt và mạnh mẽ, giúp phát triển ứng dụng streaming nhạc trở nên dễ dàng và hiệu quả. [2]

- .NET hỗ trợ đa nền tảng, cho phép ứng dụng streaming nhạc chạy trên nhiều hệ điều hành khác nhau, từ Android, Apple, Linux đến Windows. [2]

- .NET đi kèm với một bộ thư viện và API tiêu chuẩn, bao gồm các tính năng liên quan đến mạng và xử lý âm thanh, giúp cho việc phát triển ứng dụng streaming nhạc trở nên dễ dàng và linh hoạt hơn. [2]

**2.3/ Địa chỉ IP:**

- Địa chỉ IP (Internet Protocol address) là một định danh duy nhất được gán cho mỗi thiết bị kết nối vào mạng internet. Nó cho phép các thiết bị giao tiếp với nhau trên mạng bằng cách xác định địa chỉ của chúng trong mạng. Địa chỉ IP có hai phiên bản phổ biến là IPv4 và IPv6.

- IPv4 (Internet Protocol version 4) là phiên bản địa chỉ IP ban đầu và vẫn được sử dụng rộng rãi. Nó bao gồm 32 bit và được biểu diễn dưới dạng dãy số và dấu chấm (ví dụ: 192.168.0.1).

- IPv6 (Internet Protocol version 6) là phiên bản mới hơn của địa chỉ IP, với kích thước 128 bit. Nó được thiết kế để đáp ứng nhu cầu tăng của mạng internet và cung cấp một phạm vi địa chỉ lớn hơn cho các thiết bị kết nối.

- Địa chỉ IP chia thành hai loại: địa chỉ IP tĩnh và địa chỉ IP động. Địa chỉ IP tĩnh là địa chỉ cố định được gán cho một thiết bị và không thay đổi, trong khi địa chỉ IP động có thể thay đổi theo thời gian hoặc theo vùng mạng.

- Địa chỉ IP đóng vai trò quan trọng trong việc xác định và định tuyến dữ liệu trên mạng internet, giúp cho việc giao tiếp giữa các thiết bị trở nên mượt mà và hiệu quả.

- Trong ứng dụng streaming nhạc trực tuyến, địa chỉ IP (Internet Protocol address) đóng vai trò quan trọng trong việc quản lý và tối ưu hóa quá trình truyền dữ liệu âm nhạc giữa máy chủ và các thiết bị người dung.

**2.4/ Socket**

- Trong lập trình mạng, Socket là một giao diện lập trình ứng dụng (API) cho phép các ứng dụng thiết lập kết nối mạng giữa các thiết bị thông qua internet. Nó cung cấp một cách tiêu chuẩn để thiết lập kết nối, truyền và nhận dữ liệu giữa các thiết bị trên mạng, và được sử dụng để triển khai các giao thức truyền dữ liệu như TCP hoặc UDP.

- Trong lập trình ứng dụng streaming nhạc trực tuyến, việc sử dụng Socket là rất quan trọng. Thông qua Socket, ứng dụng có thể thiết lập kết nối giữa máy chủ (server) chứa dữ liệu nhạc và các thiết bị người dùng (client) để truyền dữ liệu âm nhạc giữa chúng.

**2.5/ TCP**

- Trong lập trình mạng, TCP (Transmission Control Protocol) là một trong hai giao thức chính được sử dụng để truyền dữ liệu giữa các thiết bị trên mạng internet. TCP cung cấp một cách tin cậy để truyền dữ liệu bằng cách đảm bảo rằng các gói tin được gửi từ một thiết bị sẽ được nhận đúng thứ tự và không bị mất trên đường truyền.

- TCP hoạt động dựa trên mô hình client-server, trong đó một thiết bị hoạt động như máy chủ và các thiết bị khác kết nối đến máy chủ để truyền và nhận dữ liệu. Các kết nối TCP được thiết lập thông qua việc gửi các gói tin chứa thông tin về địa chỉ IP và cổng của máy chủ và client.

- TCPClient và TCPListener là hai lớp quan trọng trong lập trình mạng sử dụng TCP trong .NET Framework.

+ TCPClient: được sử dụng để tạo và quản lý kết nối TCP tới một máy chủ (server) từ một ứng dụng client. Lớp này cung cấp các phương thức và thuộc tính để thiết lập kết nối TCP, gửi và nhận dữ liệu giữa client và server. Để sử dụng TCPClient, bạn cần chỉ định địa chỉ IP và cổng của máy chủ mà bạn muốn kết nối.

+ TCPListener: được sử dụng để tạo và quản lý một máy chủ (server) TCP, lắng nghe các yêu cầu kết nối từ các ứng dụng client. Lớp này cho phép ứng dụng server lắng nghe các kết nối đến trên một cổng nhất định và chấp nhận các kết nối mới từ các ứng dụng client. TCPListener cung cấp các phương thức để bắt đầu lắng nghe, chấp nhận kết nối và đóng kết nối. Khi một kết nối mới được chấp nhận, TCPListener trả về một đối tượng TCPClient để quản lý kết nối với client đó.

Trong ứng dụng streaming nhạc trực tuyến, TCP được sử dụng thông qua Socket để truyền dữ liệu âm nhạc giữa máy chủ chứa dữ liệu nhạc và các thiết bị người dùng (client). Socket đóng vai trò quan trọng trong việc thiết lập và quản lý kết nối giữa client và server để truyền dữ liệu âm nhạc một cách tin cậy và hiệu quả.

**2.6/ API**

- API (Application Programming Interface) là các cơ chế cho phép hai thành phần phần mềm tương tác với nhau bằng cách sử dụng một bộ định nghĩa và giao thức nhất định. [3]

- Kiến ​​trúc API thường được giải thích dựa trên khái niệm về client và server. Ứng dụng gửi yêu cầu được gọi là client, và ứng dụng gửi phản hồi được gọi là server. [3]

- Có bốn cách khác nhau mà API có thể hoạt động tùy thuộc vào thời điểm và lý do chúng được tạo:

+ API SOAP: Các API này sử dụng giao thức Simple Object Access Protocol. Client và server trao đổi thông điệp bằng cách sử dụng XML. Đây là một API ít linh hoạt hơn đã phổ biến hơn trong quá khứ. [3]

+ RPC APIs: Các API này được gọi là Remote Procedure Calls. Client hoàn thành một chức năng (hoặc thủ tục) trên server, và server gửi kết quả trở lại cho client. [3]

+ Websocket APIs: là một cách phát triển API web hiện đại khác sử dụng các đối tượng JSON để truyền dữ liệu. Một API WebSocket hỗ trợ giao tiếp hai chiều giữa ứng dụng client và server. Server có thể gửi các thông điệp gọi lại cho các client đã kết nối, làm cho nó hiệu quả hơn so với REST API. [3]

+ REST APIs (Representational State Transfer APIs) là một kiểu thiết kế API phổ biến trên web hiện nay, được xây dựng trên nguyên tắc cơ bản của REST. REST API sử dụng các HTTP methods như GET, POST, PUT, DELETE để thực hiện các thao tác CRUD (Create, Read, Update, Delete) trên tài nguyên của server. [3]

* GET: Sử dụng để lấy dữ liệu từ server.
* POST: Sử dụng để tạo mới một tài nguyên trên server.
* PUT: Sử dụng để cập nhật một tài nguyên đã tồn tại trên server hoặc tạo một tài nguyên mới nếu không tồn tại.
* DELETE: Sử dụng để xóa một tài nguyên trên server.

REST API cung cấp một cách linh hoạt và hiệu quả để tương tác với các dịch vụ web, cho phép các ứng dụng và dịch vụ tương tác với nhau một cách dễ dàng và đồng bộ.

**2.7/ HTTP (Hypertext Transfer Protocol)**

HTTP là một giao thức tiêu chuẩn được sử dụng để truyền dữ liệu trên mạng internet. Nó là cơ sở của việc truyền tải dữ liệu giữa máy chủ và trình duyệt web, cũng như giữa các ứng dụng web và máy chủ.

**Cách Hoạt Động của HTTP:**

* **Giao thức yêu cầu-phản hồi**: HTTP hoạt động dựa trên mô hình yêu cầu-phản hồi, trong đó trình duyệt hoặc ứng dụng client gửi yêu cầu đến máy chủ, và máy chủ trả lại phản hồi chứa dữ liệu tương ứng.
* **Phương thức yêu cầu (HTTP Methods)**: Có nhiều phương thức yêu cầu khác nhau trong HTTP, như **GET**, **POST**, **PUT**, **DELETE**, **PATCH**, v.v. Mỗi phương thức có một mục đích và tác dụng khác nhau.
* **URLs (Uniform Resource Locators)**: URLs xác định địa chỉ của tài nguyên trên mạng internet. Chúng được sử dụng trong yêu cầu HTTP để chỉ định tài nguyên mà client muốn truy cập.

**HTTP Trong Ứng Dụng Streaming Nhạc:**

* Trong ứng dụng streaming nhạc trực tuyến, HTTP được sử dụng để gửi yêu cầu và nhận phản hồi giữa trình duyệt hoặc ứng dụng client và máy chủ chứa dữ liệu nhạc.
* Các yêu cầu HTTP có thể được sử dụng để lấy thông tin về các bài hát, album, danh sách phát.
* Phản hồi từ máy chủ thường chứa dữ liệu nhạc được yêu cầu, và client có thể sử dụng dữ liệu này để phát nhạc trực tuyến cho người dùng.

**RESTful APIs và HTTP:**

* Trong môi trường phát triển ứng dụng web hiện đại, RESTful APIs thường được sử dụng để tạo các dịch vụ web. RESTful APIs sử dụng HTTP để truyền tải dữ liệu và sử dụng các phương thức HTTP như **GET**, **POST**, **PUT**, **DELETE** để quản lý tài nguyên.
* Các ứng dụng streaming nhạc thường cung cấp RESTful APIs để cho phép các client truy cập và tương tác với dịch vụ của họ thông qua HTTP.

HTTP đóng vai trò quan trọng trong việc truyền tải dữ liệu giữa các thành phần trong ứng dụng streaming nhạc trực tuyến. Sử dụng HTTP và RESTful APIs giúp tạo ra các ứng dụng linh hoạt, dễ mở rộng và tương tác tốt với các dịch vụ trên mạng internet.

**2.8/ API với Ngôn Ngữ Lập Trình Golang**

- Golang, hay Go, là một ngôn ngữ lập trình được phát triển bởi Google với mục tiêu đơn giản, hiệu quả và dễ đọc. Nó được thiết kế để xử lý các vấn đề phức tạp trong phát triển phần mềm, đặc biệt là trong việc xây dựng hệ thống có khả năng mở rộng và độ tin cậy cao.

- Sử Dụng Golang để Xây Dựng API:

+ Hiệu Suất: Golang được thiết kế với mục tiêu đảm bảo hiệu suất cao. Với hệ thống quản lý bộ nhớ hiệu quả và khả năng xử lý đa luồng, Go là lựa chọn lý tưởng cho việc xây dựng các API yêu cầu hiệu suất cao.

+ Xử Lý Đa Luồng (Concurrency): Golang cung cấp các công cụ như Goroutines và Channels giúp quản lý đa luồng một cách dễ dàng và hiệu quả. Điều này làm cho việc xây dựng các API đồng thời và có khả năng mở rộng trở nên dễ dàng.

+ Dễ Đọc và Dễ Học: Cú pháp rõ ràng và đơn giản của Go làm cho mã nguồn trở nên dễ đọc và dễ hiểu, giúp cho việc phát triển và bảo trì API trở nên dễ dàng hơn.

+ Tích Hợp Tốt: Golang hỗ trợ tích hợp dễ dàng với các thư viện và công nghệ khác, giúp việc phát triển API linh hoạt và dễ dàng tích hợp với các hệ thống khác.

+ Bảo Mật: Golang đi kèm với các công cụ và thư viện bảo mật mạnh mẽ, giúp bảo vệ API khỏi các lỗ hổng bảo mật phổ biến.

+ Cộng Đồng Phát Triển Mạnh Mẽ: Mặc dù không phải là ngôn ngữ phổ biến nhất, nhưng cộng đồng phát triển của Go là nhiệt tình và hỗ trợ, cung cấp nhiều tài liệu và thư viện mã nguồn mở hữu ích cho việc phát triển API.

- Xây Dựng API với Golang:

+ Golang cung cấp một số thư viện tiêu chuẩn và các framework như Gin, Echo, hoặc Beego để giúp xây dựng các API một cách nhanh chóng và linh hoạt.

+ Các API có thể được thiết kế theo mô hình RESTful để đảm bảo tính mở rộng và dễ bảo trì.

+ Golang cũng hỗ trợ việc xây dựng các dịch vụ web socket và giao thức HTTP/2, giúp tăng cường hiệu suất và khả năng tương tác trong việc truyền tải dữ liệu.

Golang là một lựa chọn lý tưởng cho việc xây dựng các API yêu cầu hiệu suất, độ tin cậy và tính mở rộng. Với hiệu suất cao, tính mở rộng dễ dàng và cú pháp đơn giản, Golang là một ngôn ngữ mạnh mẽ cho việc phát triển các dịch vụ web và API.

**2.9/ Database với SQLite 3**

- SQLite 3 là một hệ quản trị cơ sở dữ liệu quan hệ nhỏ gọn, không cần máy chủ, và có hiệu suất cao. Nó là một lựa chọn phổ biến cho các ứng dụng có nhu cầu lưu trữ dữ liệu đơn giản như ứng dụng di động, máy tính cá nhân và các dự án nhỏ.

- Đặc Điểm Của SQLite 3:

+ Không Cần Máy Chủ: SQLite 3 không yêu cầu máy chủ riêng biệt để triển khai và quản lý cơ sở dữ liệu. Điều này làm cho việc triển khai và tích hợp SQLite 3 trở nên dễ dàng và tiết kiệm chi phí.

+ Nhỏ Gọn: SQLite 3 có kích thước nhỏ gọn, chỉ cần một tập tin duy nhất để lưu trữ toàn bộ cơ sở dữ liệu. Điều này làm cho việc di chuyển, sao lưu và phục hồi dữ liệu trở nên thuận tiện.

+ Hỗ Trợ Rộng Rãi: SQLite 3 được hỗ trợ trên nhiều nền tảng, bao gồm Windows, macOS, Linux và các hệ điều hành di động như Android và iOS.

+ Hiệu Suất Cao: Mặc dù nhỏ gọn, nhưng SQLite 3 cung cấp hiệu suất cao và khả năng xử lý tốt, làm cho nó trở thành lựa chọn phù hợp cho các ứng dụng có nhu cầu xử lý dữ liệu lớn.

+ Hỗ Trợ SQL Chuẩn: SQLite 3 hỗ trợ một phạm vi rộng rãi các lệnh và chức năng SQL chuẩn, bao gồm SELECT, INSERT, UPDATE, DELETE, và JOIN, giúp cho việc thao tác với cơ sở dữ liệu trở nên linh hoạt và dễ dàng.

- Sử Dụng SQLite 3 Trong Ứng Dụng Streaming Nhạc Trực Tuyến:

+ Lưu Trữ Dữ Liệu Người Dùng: SQLite 3 có thể được sử dụng để lưu trữ thông tin về người dùng như thông tin đăng nhập, lịch sử nghe nhạc, và cài đặt cá nhân.

+ Quản Lý Danh Sách Phát: Cơ sở dữ liệu SQLite 3 có thể được sử dụng để lưu trữ và quản lý danh sách phát của người dùng, bao gồm thông tin về bài hát, album, và danh sách phát cá nhân.

+ Thống Kê Và Phân Tích Dữ Liệu: SQLite 3 có thể được sử dụng để thực hiện các truy vấn phức tạp nhằm phân tích và hiểu rõ hơn về hành vi người dùng, góp phần vào việc cải thiện trải nghiệm người dùng và tối ưu hóa dịch vụ.

Trong đồ án ứng dụng streaming nhạc trực tuyến, việc sử dụng SQLite 3 giúp quản lý và lưu trữ dữ liệu một cách hiệu quả, từ thông tin người dùng đến các danh sách phát và dữ liệu thống kê. Điều này giúp cải thiện trải nghiệm người dùng và tăng tính mở rộng của ứng dụng.

# **Tài liệu tham khảo**

|  |  |
| --- | --- |
| [1] | “snaplogic,” [Trực tuyến]. Available: https://www.snaplogic.com/glossary/network-application. [Đã truy cập 6 4 2024]. |
| [2] | Microsoft, “What is .NET | .NET,” Microsoft, 2024. [Trực tuyến]. Available: https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet.. [Đã truy cập 6 April 2024]. |
| [3] | Amazon, “What is an API?,” Amazon, [Trực tuyến]. Available: https://aws.amazon.com/what-is/api/. [Đã truy cập 7 April 2024]. |