**Công nghệ sử dụng**

1. **DataBase MySQL**

**MySQL** là [hệ quản trị cơ sở dữ liệu](https://vi.wikipedia.org/wiki/H%E1%BB%87_qu%E1%BA%A3n_tr%E1%BB%8B_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) [tự do nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. Người dùng có thể tải về MySQL miễn phí từ trang chủ. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux), [Mac OS X](https://vi.wikipedia.org/wiki/Mac_OS_X), [Unix](https://vi.wikipedia.org/wiki/Unix), [FreeBSD](https://vi.wikipedia.org/wiki/FreeBSD), [NetBSD](https://vi.wikipedia.org/w/index.php?title=NetBSD&action=edit&redlink=1), [Novell NetWare](https://vi.wikipedia.org/wiki/Novell_NetWare), [SGI Irix](https://vi.wikipedia.org/w/index.php?title=SGI_Irix&action=edit&redlink=1), [Solaris](https://vi.wikipedia.org/wiki/Solaris), [SunOS](https://vi.wikipedia.org/w/index.php?title=SunOS&action=edit&redlink=1),..

**MySQL** là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

**MySQL** được sử dụng cho việc bổ trợ NodeJs, [PHP](https://vi.wikipedia.org/wiki/PHP), [Perl](https://vi.wikipedia.org/wiki/Perl), và nhiều ngôn ngữ khác, làm nơi lưu trữ những thông tin trên các trang web viết bằng NodeJs, PHP hay Perl,...

MySQL sở hữu những tính năng đầy hữu ích dưới đây:

* Là cơ sở dữ liệu **tốc độ cao, ổn định**
* **Độ bảo mật thông tin cao**
* **Dễ sử dụng** và **có tính khả chuyển**: tuy có tính năng cao nhưng thực sự, MySQL là một hệ thống cơ sở dữ liệu rất đơn giản và ít phức tạp .
* Có thể hoạ**t động trên nhiều hệ điều hành**với lượng lớn các hàm tiện ích: MySQL rất thích hợp cho các ứng dụng có truy cập cơ sở dữ liệu trên Internet. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,..
* **Hoàn toàn miễn phí:** Là một mã nguồn mở, MySQL sử dụng GNU General Public License nên hoàn toàn miễn phí. Tuy nhiên,trong trường hợp bạn cần được hỗ trợ từ MySQL, bạn vẫn phải trả phí.
* **Cộng đồng hỗ trợ :** Vì lượng người dùng MySQL vô cùng lớn nên sự hỗ trợ từ cộng đồng này cũng lớn theo. Khi trở thành một thành viên của cộng đồng người dùng, bạn sẽ được mọi người trong cộng đồng tư vấn giải pháp họ biết cho vấn đề bạn gặp phải.

Tuy nhiên, MySQL vẫn còn một số hạn chế nhất định:

* MySQL có thể bị khai thác để chiếm quyền điều khiển.
* MySQL không được tích hợp để sử dụng cho các hệ thống lớn cần quản lý lượng dữ liệu khổng lồ. Ví dụ: hệ thống siêu thị trên toàn quốc, ngân hàng, quản lý thông tin dân số cả nước, …

1. **Ngôn ngữ lập trình JAVA**

[Java](https://techmaster.vn/khoa-hoc/25467/java-can-ban-thuc-hanh) là một ngôn ngữ lập trình phổ biến. Điều này được chứng minh thực tế trong suốt 20 năm qua.

Hai thập kỷ không phải là một thời gian ngắn cho bất kỳ một ngôn ngữ lập trình nào, và Java đã khẳng định được sức mạnh từng ngày. Dù có những lúc, Java phát triển chậm lại, nhưng nó đã thích ứng tốt. Trước đó, với thay đổi cơ bản về hình thức Enum, Generics, và autoboxing trong Java 5, cải thiện hiệu suất với Java 6, và việc Google lựa chọn ngôn ngữ Java để phát triển ứng dụng Android, Java vẫn giữ vị trí là ngôn ngữ lập trình tiên phong.

Java sở hữu những tính năng hữu ích dưới đây:

* Java rất dễ tìm hiểu
* Java là một ngôn ngữ lập trình hướng đối tượng
* Số lượng hàm dùng sẵn (API function) của Java hết sức phong phú
* Các công cụ phát triển mạnh mẽ như Eclipse, Netbeans
* Bộ sưu tập thư viện mã nguồn mở phong phú
* Hỗ trợ cộng đồng tuyệt vời
* Java là miễn phí
* Hỗ trợ tài liệu xuất sắc – Javadocs
* Java là nền tảng độc lập
* Java có mặt ở khắp mọi nơi

## Thư viện Spring Framework Boot

## [Spring Framework](https://spring.io/projects/spring-framework) cung cấp một mô hình lập trình và cấu hình toàn diện cho các ứng dụng doanh nghiệp dựa trên Java hiện đại - trên bất kỳ nền tảng nào.

## Phần Core của Spring Framework được sử dụng để xây dựng bất kỳ ứng dụng Java nào

## Hơn thế nữa, vì nó cũng cung cấp hỗ trợ cho nhiều framework như Big Data, Security, JPA, Social Integration, Web-services, v.v.. nên Spring Framework thực sự được coi là "Điện nước đầy đủ"!

## Những lý do nên sử dụng Spring

## Spring nhẹ đến bất ngờ

## **Spring được tổ chức tốt**

## **Khả năng quản lý giao dịch**

## **Khả năng kiểm thử của Spring**

## **Spring hỗ trợ Lập trình AOP**

## **Spring dễ tích hợp với các Framework khác**

## **Tính Module mạnh mẽ**

## **Spring được thiết kế tốt**

## **Spring Thân thiện với Lập trình viên**

## **Spring có API tiện lợi**

## Thực tế, theo khảo sát đầu năm 2019 của [Jetbrains](https://www.jetbrains.com/lp/devecosystem-2019/java/) tỷ lệ sử dụng Spring Framework là rất cao.

Ngoài ra, **Spring framework** cũng có một **cộng đồng rất lớn**. Họ luôn sẵn sàng cho bạn câu trả lời đúng đắn. Chắc chắn rằng, Spring framework có đầy đủ tính năng bạn cần cho dự án Java Web của bạn.

1. **Truy vấn Heibernate**

Hibernate là một thư viện ORM (Object Relational Mapping) mã nguồn mở giúp lập trình viên viết ứng dụng Java có thể map các objects (pojo) với hệ quản trị cơ sở dữ liệu quan hệ, và hỗ trợ thực hiện các khái niệm lập trình hướng đối tượng với cớ dữ liệu quan hệ.

Hibernate Framework có các lợi ích như dưới đây:

* **Mã nguồn mở và nhẹ**: Hibernate Framework là mã nguồn mở có giấy phép LGPL và nhẹ.
* **Hiệu suất nhanh**: Hiệu suất của Hibernate Framework là nhanh bởi vì bộ nhớ cache được sử dụng trong nội bộ Hibernate Framework. Có hai loại bộ nhớ cache trong Hibernate Framework, gồm bộ nhớ cache cấp một (first level cache) và bộ nhớ cache cấp hai (second level cache). Bộ nhớ cache cấp một được enable mặc định.
* **Truy vấn cơ sở dữ liệu độc lập**: HQL (Hibernate Query Language) là phiên bản hướng đối tượng của SQL. Nó tạo ra các truy vấn cơ sở dữ liệu độc lập. Vì vậy, bạn không cần phải viết các truy vấn cơ sở dữ liệu cụ thể. Trước Hibernate, nếu dự án có cơ sở dữ liệu bị thay đổi, chúng ta cần phải thay đổi truy vấn SQL dẫn đến risk và dễ gây lỗi.
* **Tạo bảng tự động**: Hibernate framework cung cấp phương tiện để tạo ra các bảng cơ sở dữ liệu một cách tự động. Vì vậy, không cần phải tốn công sức tạo ra các bảng trong cơ sở dữ liệu thủ công.
* **Đơn giản lệnh join phức tạp**: Có thể lấy dữ liệu từ nhiều bảng một cách dễ dàng với Hibernate framework.
* **Cung cấp thống kê truy vấn và trạng thái cơ sở dữ liệu**: Hibernate hỗ trợ bộ nhớ cache truy vấn và cung cấp số liệu thống kê về truy vấn và trạng thái cơ sở dữ liệu.

1. **ANGULAR JS 9**

ANGULAR JS là một **Frameworks Javascript** giúp chúng ta xây dựng ứng dụng **Web** đầy đủ tính năng từ phía **Client**. **Angular**lần đầu được phát hành bởi gã khổng lồ **Google** vào năm 2010 với phiên bản **AngularJS**, sau đó đã có chỗ đứng khá vững chắc trong một thời gian dài, phiên bản **Angular 2** phát hành năm 2016 mang đến một bước chuyển mình vượt bậc, một công cụ thực sự mạnh mẽ cho việc phát triển ứng dụng **Web** trên cả nền tảng **Mobile** và **Desktop**. Hiện nay chúng ta đang có phiên bản **Angular 9**

### - Angular giúp nâng cao năng suất của các lập trình viên.

### - Cấu trúc phát triển rõ rang

Mô hình **MVVM** cũng giúp **Angular** chiếm lợi thế trong xây dựng ứng dụng **client-side**, thường ta sẽ có 3 điều cần quan tâm chính: đó là **giao diện người dùng**, mã nguồn điều khiển giao diện và mô hình dữ liệu (**data**) cho giao diện. Angular với **MVVM** phân biệt hoàn toàn rõ ràng các yếu tố trên nhờ mô hình **MVVM**:

* Phần giao diện (**view**) được định nghĩa trong một **template** bao hàm **HTML** dành cho một **component** nhất định. **Template** có thể là toàn bộ **Layout** hoặc bất cứ mảnh ghép nào trong **Layout** đó.
* **Model** được định nghĩa như là các thuộc tính của **component class**. Có thể hiểu là dữ liệu, dựa vào đó để phần **View** sử dụng để thực thi.
* **view/model** là **class** quản lý cả **view** cũng như **model**. Là phần **code** sẽ xử lý việc truy xuất dữ liệu, đồng thời thực thi các tương tác của người dùng trên **view**.  
  Với việc ứng dụng các điểm tích cực của các thành phần trên, **Angular** khiến việc phát triển ứng dụng trở nên dễ dàng và hiệu quả hơn.

### Hỗ trợ đầy đủ tính năng điều hướng

### **Angular** giúp giảm tối đa kích thước và tăng tối đa hiệu suất của ứng dụng.

### Document và cộng đồng