



Biểu đồ triển khai trong Ngôn ngữ mô hình hóa thống nhất (UML)

Cập nhật lần cuối: 23 tháng 7 năm 2025

Biểu đồ Triển khai là một loại [Biểu đồ Cấu trúc UML](#), thể hiện việc triển khai vật lý các thành phần phần mềm trên các nút phần cứng. Biểu đồ này minh họa việc ánh xạ các thành phần phần mềm vào các tài nguyên vật lý của hệ thống, chẳng hạn như máy chủ, bộ xử lý, thiết bị lưu trữ và cơ sở hạ tầng mạng.



Mục lục

- [Biểu đồ triển khai là gì?](#)
- [Các yếu tố chính của Biểu đồ triển khai](#)
- [Các thành phần và ký hiệu trong sơ đồ triển khai](#)
- [Các trường hợp sử dụng biểu đồ triển khai](#)
- [Các bước để tạo Biểu đồ triển khai](#)
- [Các mẫu triển khai](#)
- [Ví dụ thực tế về sơ đồ triển khai](#)
- [Tích hợp Biểu đồ triển khai với các Biểu đồ UML khác](#)
- [Lợi ích của Biểu đồ triển khai](#)
- [Những thách thức của biểu đồ triển khai](#)

Biểu đồ triển khai là gì?

Biểu đồ Triển khai cho thấy cách thiết kế phần mềm chuyển thành hệ thống vật lý thực tế nơi phần mềm sẽ chạy. Biểu đồ này cho thấy vị trí các thành phần phần mềm được đặt trên các thiết bị phần cứng và cách chúng kết nối với nhau. Biểu đồ này giúp hình dung cách phần mềm sẽ hoạt động trên các thiết bị khác nhau.

Các yếu tố chính của Biểu đồ triển khai

Dưới đây là các yếu tố chính của sơ đồ triển khai:

- **Nút** : Đây là các thực thể phần cứng vật lý nơi các thành phần phần mềm được triển khai, chẳng hạn như máy chủ, máy trạm, bộ định tuyến, v.v.
- **Thành phần** : Biểu thị các mô-đun phần mềm hoặc hiện vật được triển khai trên các nút, bao gồm các tệp thực thi, thư viện, cơ sở dữ liệu và tệp cấu hình.
- **Hiện vật** : Các tệp vật lý được đặt trên các nút thể hiện việc triển khai thực tế các thành phần phần mềm. Chúng có thể bao gồm các tệp thực thi, tập lệnh, cơ sở dữ liệu, v.v.
- **Phụ thuộc** : Chúng biểu thị các mối quan hệ hoặc kết nối giữa các nút và thành phần, làm nổi bật các đường dẫn truyền thông, hạn chế triển khai

[Hướng dẫn thiết kế hệ thống](#)

[Thiết kế hệ thống là gì](#)

[Vòng đời thiết kế hệ thống](#)

[Thiết kế cấp cao HLD](#)

[Thiết kế cấp thấp LLD](#)

[Mẫu thiết kế](#)

[Biểu đồ UML](#)

[Hướng dẫn phỏng](#)

[Đăng nhập](#)

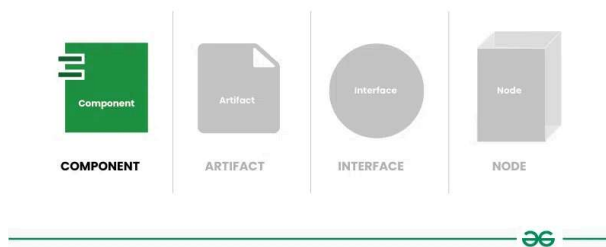
- **Liên kết** : Hiển thị mối quan hệ giữa các nút và thành phần, biểu thị rằng một thành phần được triển khai trên một nút vật lý.
- **Đặc tả triển khai** : Phần này nêu rõ thiết lập và đặc điểm của các nút và thành phần, bao gồm giao thức truyền thông.
- **Đường dẫn truyền thông** : Biểu thị các kênh hoặc kết nối tạo điều kiện thuận lợi cho việc truyền tải dữ liệu, giao thức truyền thông, v.v.

Các thành phần và ký hiệu trong sơ đồ triển khai

Dưới đây là các thành phần và ký hiệu của chúng trong sơ đồ triển khai:

1. Thành phần

Component | Unified Modeling Language(UML)

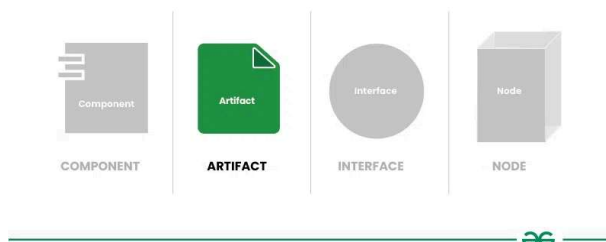


Một thành phần đại diện cho một phần mô-đun và có thể tái sử dụng của hệ thống, thường được triển khai dưới dạng một mô-đun, lớp hoặc gói phần mềm. Nó đóng gói hành vi và dữ liệu của chính nó và có thể được triển khai độc lập.

Thường được biểu diễn dưới dạng **hình chữ nhật với hai hình chữ nhật nhỏ hơn nhô ra từ hai bên**, biểu thị các cổng kết nối. Tên của thành phần được viết bên trong hình chữ nhật.

2. Hiện vật

Artifact | Unified Modeling Language(UML)

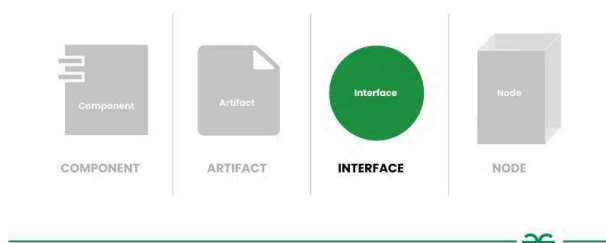


Một hiện vật đại diện cho một phần thông tin hoặc dữ liệu vật lý được sử dụng hoặc tạo ra trong quá trình phát triển phần mềm. Nó có thể bao gồm các tệp mã nguồn, tệp thực thi, tài liệu, thư viện, tệp cấu hình hoặc bất kỳ mục nào khác.

Thông thường được biểu diễn dưới dạng **hình chữ nhật có góc gấp**, được dán nhãn tên hiện vật. Hiện vật cũng có thể bao gồm thông tin bổ sung, chẳng hạn như phần mở rộng tệp hoặc phiên bản.

3. Giao diện

Interface | Unified Modeling Language(UML)

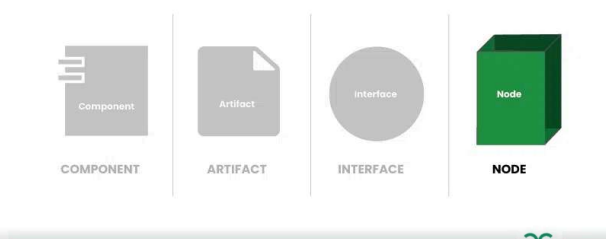


Giao diện định nghĩa một hợp đồng, trong đó chỉ rõ các phương thức hoặc thao tác mà một thành phần phải triển khai. Nó đại diện cho một điểm tương tác giữa các thành phần hoặc hệ thống con khác nhau.

Được biểu diễn dưới dạng **hình tròn hoặc hình elip**, được gắn nhãn tên giao diện. Giao diện cũng có thể bao gồm giao diện được cung cấp và giao diện bắt buộc, được biểu thị bằng ký hiệu "+" và "-".

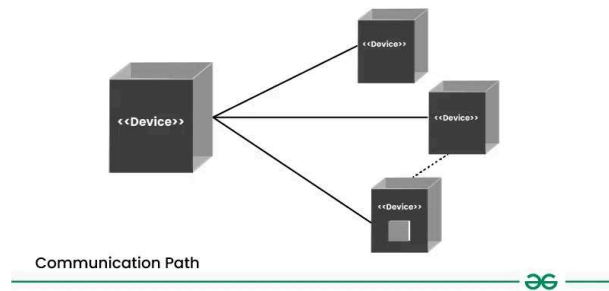
4. Nút

Node | Unified Modeling Language(UML)



Được biểu diễn dưới dạng một hộp có các góc bo tròn, thường được dán nhãn bằng tên của nút. Các nút cũng có thể bao gồm các nút lồng nhau để biểu diễn các cấu trúc phân cấp.

5. Đường truyền thông tin



Đường thẳng biểu thị giao tiếp giữa hai nút thiết bị. Đường nét đứt trong sơ đồ triển khai thể hiện mối quan hệ hoặc sự phụ thuộc giữa các phần tử, cho biết một phần tử có liên quan hoặc phụ thuộc vào phần tử khác.

Các trường hợp sử dụng biểu đồ triển khai

Dưới đây là các trường hợp sử dụng sơ đồ triển khai:

- Biểu đồ triển khai giúp lập kế hoạch cách thiết lập hệ thống phần mềm trên các thiết bị khác nhau.
- Họ giúp thiết kế phần cứng cần thiết để hỗ trợ phần mềm. Bằng cách chỉ ra phần mềm nào nằm ở đâu, họ giúp quyết định thiết bị và mạng nào là cần thiết.
- Biểu đồ triển khai đảm bảo mỗi phần của phần mềm có đủ tài nguyên, như bộ nhớ hoặc sức mạnh xử lý, để chạy tốt.
- Chúng cho thấy các phần khác nhau của phần mềm phụ thuộc vào nhau và vào phần cứng như thế nào.
- Bằng cách xem cách mọi thứ được thiết lập, các nhóm có thể tìm ra cách để phần mềm chạy nhanh hơn và mượt mà hơn.

Các bước để tạo Biểu đồ triển khai

Dưới đây là các bước chính để tạo sơ đồ triển khai:

- **Bước 1: Xác định các thành phần:** Liệt kê tất cả các bộ phận phần mềm và thiết bị phần cứng sẽ có trong sơ đồ triển khai.
- **Bước 2: Hiểu các mối quan hệ:** Tìm hiểu cách các bộ phận này kết nối và hoạt động cùng nhau.
- **Bước 3: Thu thập các yêu cầu:** Thu thập thông tin chi tiết về phần cứng, thiết lập mạng và bất kỳ quy tắc đặc biệt nào cho việc triển khai.
- **Bước 4: Vẽ các nút và thành phần:** Bắt đầu bằng cách vẽ các thiết bị phần cứng (nút) và các bộ phận phần mềm (thành phần) bằng cách sử dụng các ký hiệu chuẩn, trước tiên hãy ứng biến và vẽ bản cuối cùng.
- **Bước 5: Kết nối các nút và thành phần:** Sử dụng các đường thẳng hoặc mũi tên để hiển thị cách liên kết các nút và thành phần.
- **Bước 6: Thêm chi tiết:** Ghi nhãn mọi thứ rõ ràng và bao gồm bất kỳ thông tin bổ sung nào, như thông số kỹ thuật phần cứng hoặc giao thức truyền thông.
- **Bước 7: Ghi chép:** Ghi lại mọi quyết định hoặc giả định quan trọng được đưa ra khi tạo sơ đồ.

Các mẫu triển khai

Các mẫu triển khai là các phương pháp chuẩn hóa để cài đặt phần mềm hiệu quả trên cơ sở hạ tầng phần cứng. Chúng cung cấp hướng dẫn để tổ chức và triển khai các thành phần phần mềm, giải quyết các thách thức như [khả năng mở rộng](#), [độ tin cậy](#) và hiệu suất.

- **Triển khai máy khách-máy chủ:** Minh họa kết nối giữa các ứng dụng máy khách và các nút máy chủ trong kiến trúc máy khách-máy chủ.
- **Kiến trúc ba tầng:** Hiển thị sự phân bố của các thành phần trình bày, logic ứng dụng và lưu trữ dữ liệu trên các nút khác nhau.
- **Kiến trúc vi dịch vụ:** Mô tả cách từng vi dịch vụ được triển khai trên các nút hoặc vùng chứa riêng biệt.
- **Container hóa:** Hiển thị việc triển khai các container khác nhau trên máy chủ bằng các công nghệ như Docker.
- **Triển khai đám mây:** Minh họa sự phân bố các thành phần trên nhiều dịch vụ đám mây hoặc khu vực khác nhau trong môi trường đám mây.

Ví dụ thực tế về sơ đồ triển khai

Dưới đây là các ví dụ thực tế về sơ đồ triển khai:

Ví dụ 1:

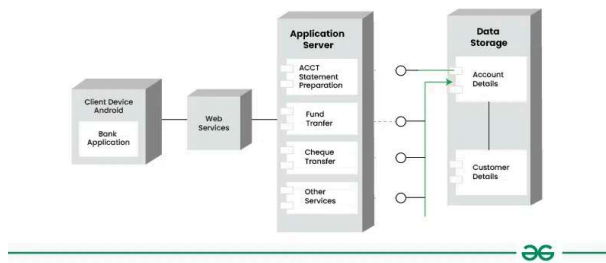
Sơ đồ triển khai dịch vụ ngân hàng di động Android.

Trong ví dụ này, một nút đại diện cho thiết bị Android của khách hàng. Các thành phần đại diện cho phần mềm được cài đặt trên các thiết bị này, với ứng dụng ngân hàng là thành phần cụ thể trên thiết bị Android.

- Sơ đồ cũng cho thấy cách người dùng kết nối với máy chủ ngân hàng thông qua web.
- Điều này có nghĩa là người dùng mở ứng dụng ngân hàng trên thiết bị Android của họ, sau đó ứng dụng sẽ kết nối với máy chủ ứng dụng trực tuyến để thực hiện các tác vụ như kiểm tra số dư tài khoản hoặc chuyển tiền.

Nhìn chung, sơ đồ triển khai minh họa trực quan cách các thành phần phần mềm được thiết lập trên các nút phần cứng và cách chúng tương tác để

Deployment Diagram for Mobile Banking Android Services

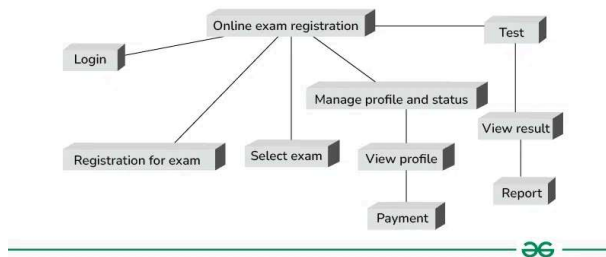


Ví dụ 2:

Sơ đồ triển khai cho hệ thống đăng ký thi trực tuyến.

Quy trình đăng ký thi trực tuyến thường bao gồm một loạt các tính năng được thiết kế để đơn giản hóa và hợp lý hóa trải nghiệm đăng ký cho người dùng. Sau khi đăng nhập hoặc đăng ký tài khoản, người dùng có thể duyệt và chọn kỳ thi mong muốn từ danh sách các tùy chọn có sẵn.

Deployment Diagram For Online Exam Registration System



Dưới đây là lời giải thích cho ví dụ trên:

- **Đăng nhập hoặc Đăng ký** : Người dùng bắt đầu bằng cách nhập thông tin đăng nhập để truy cập hệ thống đăng ký thi. Người dùng mới phải đăng ký bằng cách cung cấp thông tin như tên, email và số điện thoại liên lạc.
- **Chọn Kỳ thi** : Sau khi đăng nhập hoặc đăng ký, người dùng có thể chọn kỳ thi mình muốn đăng ký. Việc này bao gồm việc chọn từ các kỳ thi có sẵn, kiểm tra ngày và địa điểm, và đảm bảo đáp ứng các yêu cầu về điều kiện dự thi.
- **Quản lý hồ sơ và trạng thái** : Người dùng có thể cập nhật hồ sơ, kiểm tra trạng thái đăng ký và in biên lai điện tử.
- **Xem hồ sơ** : Người dùng có thể truy cập thông tin đã đăng ký của mình để xác nhận tính chính xác, cập nhật thông tin liên hệ, in biên lai điện tử và lưu hồ sơ.
- **Thanh toán** : Sau khi chọn kỳ thi, người dùng sẽ đến trang thanh toán để thanh toán lệ phí thi bằng các tùy chọn như thẻ tín dụng/thẻ ghi nợ, ngân hàng trực tuyến hoặc ví điện tử.
- **Kiểm tra** : Sau khi đăng ký và thanh toán, người dùng có thể truy cập các bài kiểm tra mẫu hoặc bài tập thực hành, nếu có, để chuẩn bị cho kỳ thi.
- **Xem kết quả** : Sau khi kỳ thi kết thúc, người dùng có thể đăng nhập vào tài khoản của mình để kiểm tra kết quả.
- **Báo cáo** : Người dùng có thể nhận được nhiều báo cáo khác nhau, chẳng hạn như bảng điểm, phiếu trả lời hoặc biểu mẫu phản hồi, tùy thuộc vào kỳ thi và chính sách của hội đồng thi.

Tích hợp Biểu đồ triển khai với các Biểu đồ UML khác

Việc tích hợp Biểu đồ triển khai với các biểu đồ UML khác giúp cung cấp cái nhìn toàn diện về hệ thống, thể hiện cả cấu trúc logic (như được mô tả trong các biểu đồ UML khác) và việc triển khai vật lý các thành phần hệ thống.

- **Biểu đồ Trường hợp Sử dụng** : Biểu đồ triển khai có thể liên quan đến biểu đồ trường hợp sử dụng để hiển thị các nút phần cứng nào tham gia vào từng trường hợp sử dụng. Điều này giúp hiểu rõ cơ sở hạ tầng vật lý cần thiết để hỗ trợ các trường hợp sử dụng khác nhau.
- **Biểu đồ Lớp** : Biểu đồ triển khai có thể được liên kết với biểu đồ lớp để hiển thị cách các lớp được phân bổ trên các nút khác nhau. Điều này giúp hiểu được sự phân bố của các thành phần phần mềm trong hệ thống.
- **Biểu đồ trình tự** : Biểu đồ triển khai có thể được kết nối với biểu đồ trình tự để minh họa cách các thông điệp được truyền tải giữa các đối tượng trong các nút khác nhau. Điều này giúp hiểu rõ hơn về giao tiếp mạng giữa các phần khác nhau của hệ thống.
- **Sơ đồ Thành phần** : Sơ đồ triển khai có thể được liên kết với sơ đồ thành phần để hiển thị cách các thành phần được triển khai trên các nút. Điều này giúp hiểu được sự phân bố vật lý của các thành phần trong hệ thống.
- **Biểu đồ hoạt động** : Biểu đồ triển khai có thể liên quan đến biểu đồ hoạt động để hiển thị cách các hoạt động được phân bổ trên các nút.

Lợi ích của Biểu đồ triển khai

Dưới đây là những lợi ích của sơ đồ triển khai:

- Biểu đồ triển khai cung cấp hình ảnh rõ ràng về cách các bộ phận phần mềm được đặt trên phần cứng, giúp dễ hiểu.
- Họ giúp các nhóm thảo luận về cách thiết lập hệ thống, giúp việc thảo luận và quyết định các chiến lược triển khai trở nên dễ dàng hơn.
- Biểu đồ triển khai hỗ trợ lập kế hoạch và quản lý quy trình triển khai, đảm bảo tài nguyên được sử dụng hiệu quả và đáp ứng các yêu cầu của hệ thống.

Dưới đây là những thách thức của sơ đồ triển khai:

- Sơ đồ triển khai có thể trở nên phức tạp, đặc biệt là trong các hệ thống lớn với nhiều bộ phận. Việc quản lý sự phức tạp này rất khó khăn.
- Việc tạo và hiểu sơ đồ triển khai cần có kiến thức về ký hiệu UML mà không phải ai cũng có.
- Việc cập nhật sơ đồ triển khai khi hệ thống thay đổi tốn nhiều thời gian và công sức.
- Biểu đồ triển khai có thể không hiển thị cách mọi thứ thay đổi theo thời gian hoặc trong quá trình sử dụng thực tế.
- Việc tạo sơ đồ triển khai thường cần rất nhiều người cùng làm việc, điều này có thể khá phức tạp.

Bình luận

Thông tin thêm

Quảng cáo với chúng tôi

Bài viết tiếp theo

Biểu đồ gói – Ngôn ngữ mô hình hóa thống nhất (UML)

Đọc tương tự

Biểu đồ UML là gì
Biểu đồ Ngôn ngữ mô hình hóa thống nhất (UML) Ngôn ngữ Mô hình Thống nhất (UML) là một ngôn ngữ mô hình hóa đa năng. Mục đích chính của UML là xác định một phương pháp chuẩn để trực quan hóa cách thức thiết kế một hệ thống. Nó khá giống với các bản thiết kế được sử dụng trong các lĩnh vực kỹ thuật khác. UML không phải là một ngôn ngữ lập trình, mà là một phương pháp trực quan. Đọc 14 phút
UML dạng đầy đủ UML, tên đầy đủ là "Ngôn ngữ Mô hình Thống nhất" (Unified Modeling Language). Đây là một ngôn ngữ mô hình hóa đa năng. Mục đích chính của UML là xác định một phương pháp chuẩn để trực quan hóa cách thức một hệ thống được thiết kế. Nó khá giống với các bản thiết kế được sử dụng trong các lĩnh vực kỹ thuật khác. UML không phải là một ngôn ngữ lập trình, mà là... Đọc 3 phút
Sơ đồ cấu trúc
Biểu đồ lớp Ngôn ngữ mô hình hóa thống nhất (UML) Biểu đồ lớp UML là một công cụ trực quan thể hiện cấu trúc của một hệ thống bằng cách hiển thị các lớp, thuộc tính, phương thức và mối quan hệ giữa chúng. Nó giúp mọi người tham gia vào dự án - chẳng hạn như nhà phát triển và nhà thiết kế - hiểu cách hệ thống được tổ chức và cách các thành phần của nó tương tác. Đọc 12 phút
Biểu đồ đối tượng Ngôn ngữ mô hình hóa thống nhất (UML) Biểu đồ đối tượng là một biểu diễn trực quan trong UML (Ngôn ngữ mô hình hóa thống nhất) minh họa các thể hiện của các lớp và mối quan hệ của chúng trong một hệ thống tại một thời điểm cụ thể. Chúng hiển thị các đối tượng, thuộc tính của chúng và các liên kết giữa chúng, cung cấp một cái nhìn tổng quan về cấu trúc của hệ thống. 8 phút đọc
Biểu đồ triển khai trong Ngôn ngữ mô hình hóa thống nhất (UML) Biểu đồ Triển khai là một loại Biểu đồ Cấu trúc UML thể hiện việc triển khai vật lý các thành phần phần mềm trên các nút phần cứng. Nó minh họa việc ánh xạ các thành phần phần mềm lên các tài nguyên vật lý của hệ thống, chẳng hạn như máy chủ, bộ xử lý, thiết bị lưu trữ và cơ sở hạ tầng mạng. 8 phút đọc
Biểu đồ gói – Ngôn ngữ mô hình hóa thống nhất (UML) Biểu đồ gói là một loại biểu đồ cấu trúc trong UML (Ngôn ngữ mô hình hóa thống nhất) dùng để tổ chức và nhóm các lớp và thành phần liên quan thành các gói. Nó thể hiện trực quan các mối quan hệ phụ thuộc và mối quan hệ giữa các gói này, giúp minh họa cách các phần khác nhau của hệ thống tương tác với nhau. 7 phút đọc
Biểu đồ hành vi
Biểu đồ hành vi Ngôn ngữ mô hình thống nhất (UML) Các ứng dụng phức tạp cần sự hợp tác và lập kế hoạch từ nhiều nhóm, do đó cần một phương thức giao tiếp rõ ràng và súc tích giữa các nhóm. Vì vậy, UML trở nên thiết yếu để giao tiếp với những người không phải lập trình viên về các yêu cầu, chức năng và quy trình thiết yếu của hệ thống. UML được liên kết với 7 phút đọc

 **GeeksforGeeks**
Địa chỉ Công ty & Truyền thông:
A-143, Tầng 7, Tòa nhà Sovereign
Corporate, Khu 136, Noida, Uttar
Pradesh (201305)

Địa chỉ đăng ký:
K 061, Tháp K, Căn hộ Gulshan Vivante,
Khu 137, Noida, Gautam Buddh Nagar,
Uttar Pradesh, 201305



Công ty

Giới thiệu về chúng tôi
Hợp pháp
Chính sách bảo mật
Trong phương tiện truyền thông
Liên hệ với chúng tôi
Quảng cáo với chúng tôi
Giải pháp doanh nghiệp GFG
Chương trình đào tạo thực tập

DSA

Cấu trúc dữ liệu
Thuật toán
DSA dành cho người mới bắt đầu
Các vấn đề cơ bản của DSA
Lộ trình DSA
100 vấn đề phỏng vấn DSA hàng đầu
Lộ trình DSA của Sandeep Jain
Tất cả các bảng gian lận

Công nghệ Web

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
Khởi động
Thiết kế web

Khoa học máy tính

Hệ điều hành
Mạng máy tính
Hệ thống quản lý cơ sở dữ liệu
Kỹ thuật phần mềm
Thiết kế logic số
Toán Kỹ thuật
Phát triển phần mềm
Kiểm thử phần mềm

Thiết kế hệ thống

Thiết kế cấp cao
Thiết kế cấp thấp
Biểu đồ UML
Hướng dẫn phỏng vấn
Mẫu thiết kế
OOAD
Trại huấn luyện thiết kế hệ thống
Câu hỏi phỏng vấn

Môn học ở trường

Toán học
Vật lý
Hoá học
Sinh vật học
Khoa học xã hội
Ngữ pháp tiếng Anh
Thương mại
GK thể giới

Ngôn ngữ

Trần
Java
C++
PHP
GoLang
SQL
Ngôn ngữ R
Hướng dẫn Android
Lưu trữ hướng dẫn

Khoa học dữ liệu và ML

Khoa học dữ liệu với Python
Khoa học dữ liệu cho người mới bắt đầu
Học máy
Toán ML
Hình dung dữ liệu
Gấu trúc
NumPy
Ngôn ngữ lập trình ngôn ngữ
Học sâu

Hướng dẫn Python

Ví dụ về lập trình Python
Dự án Python
Python Tkinter
Thu thập dữ liệu web bằng Python
Hướng dẫn OpenCV
Câu hỏi phỏng vấn Python
Django

DevOps

Git
Linux
AWS
Docker
Kubernetes
Xanh lam
GCP
Lộ trình DevOps

Chuẩn bị phỏng vấn

Lập trình cạnh tranh
DS hoặc Algo hàng đầu cho CP
Quy trình tuyển dụng theo công ty
Chuẩn bị theo từng công ty
Chuẩn bị năng khiếu
Câu đố

Video GeeksforGeeks

DSA
Trần
Java
C++
Phát triển web
Khoa học dữ liệu
Môn Khoa học máy tính