**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Thành viên hội đồng**

*(Ký tên và ghi rõ họ tên)*

LỜI CẢM ƠN

Để hoàn thành đồ án tốt nghiệp với đề tài “Xây dựng website bán trà sữa bằng ASP.NET”, em đã nhận được rất nhiều sự giúp đỡ, động viên và hướng dẫn tận tình từ thầy cô, bạn bè, đồng nghiệp cũng như sự hỗ trợ từ nhiều nguồn tài liệu.

Trước tiên, em xin bày tỏ lòng biết ơn sâu sắc đến Thạc sĩ Đoàn Phước Miền, người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình thực hiện đồ án. Những ý kiến đóng góp quý báu và sự hướng dẫn của thầy đã giúp em hoàn thiện đề tài này một cách tốt nhất.

Em cũng xin gửi lời cảm ơn chân thành đến các thầy cô giảng viên Khoa Công nghệ Thông tin, trường Đại học Trà Vinh, cùng hệ thống học tập từ xa TVU Onschool, những người đã trang bị cho em những kiến thức nền tảng quan trọng trong suốt quá trình học tập tại trường.

Đặc biệt, em xin cảm ơn các cô Phạm Ngọc Hiền và Vũ Thị Ngọc Huyền, những người đã tận tâm hỗ trợ và quản lý việc học tập của em, giúp em vượt qua nhiều khó khăn trong quá trình học tập và nghiên cứu.

Em cũng xin cảm ơn các bạn cùng lớp Khoa Công nghệ thông tin, những người đã luôn đồng hành, chia sẻ kinh nghiệm và hỗ trợ em trong suốt quá trình học tập và thực hiện đồ án.

Bên cạnh đó, em xin gửi lời tri ân đến các anh chị đồng nghiệp tại công ty cổ phần IVS , những người đã không ngần ngại chia sẻ kiến thức và kinh nghiệm thực tế quý báu, giúp em vận dụng hiệu quả vào đề tài này.

Cuối cùng, em xin cảm ơn các tác giả sách báo, các tài liệu trên internet và anh chị đi trước đã nghiên cứu và đúc kết kinh nghiệm, cung cấp nguồn tư liệu tham khảo vô cùng hữu ích để em hoàn thiện đồ án.

Dù đã nỗ lực hết sức, nhưng với thời gian và kinh nghiệm còn hạn chế, đồ án chắc chắn không tránh khỏi những thiếu sót. Em rất mong nhận được sự đóng góp ý kiến từ quý thầy cô và mọi người để đồ án của em được hoàn thiện hơn.

Một lần nữa, em xin chân thành cảm ơn!

Trà Vinh, ngày tháng 03 năm 2025

Sinh viên thực hiện

*(Ký và ghi rõ họ tên)*

**Lâm Quốc Đạt**

Mục lục

*Trang*

[1](#_Toc188826862)

[2](#_Toc188826863)

[LỜI CẢM ƠN 3](#_Toc188826864)

[Mục lục 4](#_Toc188826865)

[DANH SÁCH HÌNH ảnh 6](#_Toc188826866)

[DANH SÁCH BẢNG biểu 7](#_Toc188826867)

[phần mở đầu 8](#_Toc188826868)

[CHƯƠNG 1 10](#_Toc188826869)

[TÌM HIỂU VỀ ASP.NET CORE VÀ ENTITY FRAMEWORK 10](#_Toc188826870)

[1. Asp.Net Core. 10](#_Toc188826871)

[1.1 Khái niệm về Asp.Net Core 10](#_Toc188826872)

[1.2 Tổng quát về Asp.Net Core 10](#_Toc188826873)

[2. Entity Framework Core. 12](#_Toc188826874)

[2.1. Khái niệm về Entity Framework Core 12](#_Toc188826875)

[2.2 ORM là gì? 13](#_Toc188826876)

[2.3 Cấu trúc của Entity Framework Core 13](#_Toc188826877)

[13](#_Toc188826878)

[CHƯƠNG 2 (Style “Cap 1\_Chuong”) 14](#_Toc188826879)

[Tựa đề chương 2 (Style “Cap 1”) 14](#_Toc188826880)

[2.1. Mục lục cấp 2 (Style “Cap 2”) 14](#_Toc188826881)

[2.2. Mục lục cấp 2 (Style “Cap 2”) 14](#_Toc188826882)

[CHƯƠNG 3 (Style “Cap 1\_Chuong”) 15](#_Toc188826883)

[Tựa đề chương 3 (Style “Cap 1”) 15](#_Toc188826884)

[3.1. Mục lục cấp 2 (Style “Cap 2”) 15](#_Toc188826885)

[3.2. Mục lục cấp 2 (Style “Cap 2”) 15](#_Toc188826886)

[CHƯƠNG 4 (Style “Cap 1\_Chuong”) 16](#_Toc188826887)

[Tựa đề chương 4 (Style “Cap 1”) 16](#_Toc188826888)

[4.1. Mục lục cấp 2 (Style “Cap 2”) 16](#_Toc188826889)

[4.2. Mục lục cấp 2 (Style “Cap 2”) 16](#_Toc188826890)

[CHƯƠNG 5 (Style “Cap 1\_Chuong”) 17](#_Toc188826891)

[Tựa đề chương 2 (Style “Cap 1”) 17](#_Toc188826892)

[5.1. Mục lục cấp 2 (Style “Cap 2”) 17](#_Toc188826893)

[5.2. Mục lục cấp 2 (Style “Cap 2”) 17](#_Toc188826894)

[CHƯƠNG 6 (Style “Cap 1\_Chuong”) 18](#_Toc188826895)

[KẾT LUẬN VÀ KIẾN NGHỊ (Style “Cap 1”) 18](#_Toc188826896)

[6.1. Kết luận (Style “Cap 2”) 18](#_Toc188826897)

[6.2. Kiến nghị (Style “Cap 2”) 18](#_Toc188826898)

[TÀI LIỆU THAM KHẢO (Style “Cap 1”) 20](#_Toc188826899)

DANH SÁCH HÌNH ảnh

*Trang*

[Hình 2.3: Entity Framework Core 13](#_Toc188826939)

[Hình 2.1: Nội dung hình tô đen thứ tự, dùng style “Hinh” 14](#_Toc188826940)

[Hình 3.1: Nội dung hình tô đen thứ tự, dùng style “Hinh” 15](#_Toc188826941)

[Hình 4.1: Nội dung hình tô đen thứ tự, dùng style “Hinh” 16](#_Toc188826942)

[Hình 5.1: Nội dung hình tô đen thứ tự, dùng style “Hinh” 17](#_Toc188826943)

DANH SÁCH BẢNG biểu

*Trang*

[Bảng 1.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”. 1](#_Toc8022280)

[Bảng 2.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”. 2](#_Toc8022281)

[Bảng 3.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”. 3](#_Toc8022282)

[Bảng 4.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”. 4](#_Toc8022283)

[Bảng 5.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”. 5](#_Toc8022284)

phần mở đầu

Ngày nay, ứng dụng công nghệ thông tin phát triển mạnh mẽ không ngừng nhằm đáp ứng nhu cầu người sử dụng với việc mạng internet ngày càng trở nên một công cụ không thể thiếu đối với cuộc sống hiện nay. Để việc mua sắm online trở nên dễ dàng thường thì các công ty, cửa hàng, shop,… tạo ra website riêng của mình để khách hàng có thể tiếp cận được với các thông tin mặt hàng của mình mà từ đó thì cũng dần làm việc thiết kế website ngày càng trở nên phổ biến và phát triển mạnh mẽ hơn.

Đối với việc mua sắm online ngày càng phổ biến như hiện nay thì việc xuất hiện nhiều trang website thương mại điện tử . Đó cũng là lý do mà em chọn đề tài này “**XÂY DỰNG WEBSITE BÁN TRÀ SỮA BẰNG CÔNG NGHỆ ASP.NET**” bản thân em cũng thích sử dụng chúng , chúng là những trang website giúp ta thuận tiện trong việc mua bán hàng hóa trở nên đơn giản chỉ cần ngồi nhà mua hàng mà không cần phải đi đến nơi bán.

Mục đích chọn đề tài này là do em muốn thử sức tìm tòi công nghệ mới và muốn tạo ra 1 sản phẩm thực tế theo thị trường hiện nay .

Để thiết kế một trang website thì cần phải có khá nhiều bước thực hiện để tạo nên một trang hoàn chỉnh. Mục tiêu đặt ra lúc này là tạo ra những trang có nội dung liên quan, những màu sắc, hiệu ứng phù hợp với từng trang làm cho khách hàng thấy được vẽ đẹp của trang website, gây nên những ấn tượng đối với khách hàng để họ tiếp tục sử dụng những sản phẩm khác của shop. Đề tài thực hiện gồm 3 chương:

**Chương 1:** **Tìm hiểu ASP.NET CORE và các framework hỗ trợ cho project.**

* + - Bước đầu tiên để làm một website thì chúng ta cần phải hiểu được bản chất ngôn ngữ mình dung để lập trình. Ở chương này đề tài tập trung giới thiệu về ASP.NET CORE và các framework hỗ trợ cho project.

**Chương 2**: **Mô tả chi tiết chu trình phát triển phần mềm**

* + - Nghiên cứu sơ bộ, thu thập yêu cầu hệ thống
    - Phân tích yêu cầu
    - Thiết kế hệ thống
    - Xây dựng phần mềm
    - Thử nghiệm hệ thống
    - Triển khai và Bảo trì hệ thống

**Kết luận.**

Trà Vinh, ngày tháng năm 2025

Sinh viên thực hiện

*(Ký và ghi rõ họ tên)*

**Lâm Quốc Đạt**

CHƯƠNG 1

TÌM HIỂU VỀ ASP.NET CORE VÀ ENTITY FRAMEWORK

1. Asp.Net Core.

1.1 Khái niệm về Asp.Net Core

**- ASP.NET Core** là một open-source mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile.

- Ứng dụng ASP.NET Core có thể chạy trên .NET Core hoặc trên phiên bản đầy đủ của .NET Framework. Nó được thiết kế để cung cấp và tối ưu development framework cho những dụng cái mà được triển khai trên đám mây (clound) hoặc chạy on-promise.

1.2 Tổng quát về Asp.Net Core

- Nó bao gồm các thành phần theo hướng module nhằm tối thiểu tài nguyên và chi phí phát triển, như vậy bạn giữ lại được sự mềm giẻo trong việc xây dựng giải pháp của bạn. Bạn có thể phát triển và chạy những ứng dụng ASP.NET Core đa nền tảng trên Windows, Mac và Linux.

- Đồng thời nó đã trở thành một mã nguồn mở. Đây là một thay đổi rất lớn và theo mình là quan trọng nhất của ASP.NET Core. Điều mà trước đây khó có một lập trình viên nào có thể nghĩ đến. Có lẽ đó cũng là một xu thế mà các ngôn ngữ lập trình hiện nay đang hướng tới.

- Bản phát hành đầu tiên của ASP.NET đã xuất hiện cách đây 15 năm trước, nó là một phần của .NET Framework. Từ đó, hàng triệu lập trình viên đã sử dụng nó để xây dựng những ứng dụng web tuyệt vời, và trên những năm đó Microsoft đã phát triển thêm nhiều tính năng mới.

- ASP.NET Core có một số thay đổi kiến trúc lớn, đó là kết quả của việc học hỏi rất nhiều từ các framework module hóa khác. ASP.NET Core không còn dựa trên System.Web.dll nữa. Nó được dựa trên một tập hợp các gói, các module hay cũng được gọi là các Nuget packages. Điều này cho phép bạn tối ưu ứng dụng của bạn để chỉ bao gồm những packages nào cần thiết. Lợi ích của nó là giúp cho ứng dụng nhỏ hơn, bảo mật chặt chẽ hơn, giảm sự phức tạp, tối ưu hiệu suất hoạt động và giảm chi phí, thời gian cho việc phát triển.

- Với ASP.NET Core bạn đạt được những nền tảng cải tiến dưới đây:

+ Hợp nhất việc xây dựng web UI và web APIs

+ Tích hợp những client-side frameworks hiện đại và những luồng phát triển

+ Hệ thống cấu hình dựa trên môi trường đám mây thật sự

+ Dependency injection được xây dựng sẵn

+ HTTP request được tối ưu nhẹ hơn

+ Có thể host trên IIS hoặc self-host trong process của riêng bạn

+ Được xây dựng trên .NET Core, hỗ trợ thực sự app versioning

+ Chuyển các thực thể, thành phần, module như những NuGet packages

+ Những công cụ mới để đơn giản hóa quá trình phát triển web hiện đại

+ Xây dựng và chạy đa nền tảng(Windows, Mac và Linux)

+ Mã nguồn mở và tập trung vào cộng đồng

- Xây dựng web UI và web API sử dụng ASP.NET CORE MVC:

+ Tạo ra các ứng dụng web có thể testing theo mô hình MVC(Model-View-Controller)

+ Xây dựng HTTP services cái mà hỗ trợ nhiều định dạng và có đầy đủ hỗ trợ cho nội dung của dữ liệu trả về

+ Razor cung cấp một ngôn ngữ hiệu quả để tạo Views

+ Tag Helper cho phép code server side tham gia vào việc tạo và render phần tử HTML

+ Model Binding tự động ánh xạ dữ liệu từ HTTP request tới tham số của method action

+ Model Validation tự động thực hiện validate client và server

- ASP.NET Core được thiết kế để tích hợp một cách liên tục với nhiều client-side frameworks, bao gồm AngularJS, KnockoutJS và Bootstrap. Về phần này tôi sẽ giới thiệu với các bạn ở những bài viết sau.

2. Entity Framework Core.

2.1. Khái niệm về Entity Framework Core

- Entity Framework Core là một Object Relational Mapper(ORM) một công cụ mapping giữa các objects trong ứng dụng với các tables và columns của cơ sở dữ liệu quan hệ.

- Entity Framework Core là một Open source ORM Framework.

- Một ORM có thể tạo kết nối đến một Relational database, thực thi cậu lệnh và trả về kết quả là các objects trong ứng dụng.

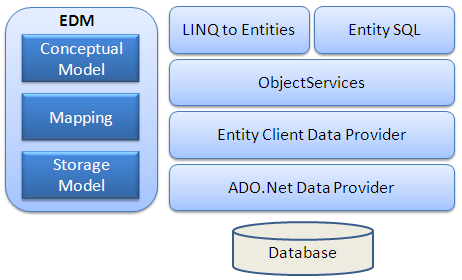
- Một ORM sẽ theo dõi sự thay đổi dữ liệu,trạng thái của các objects trả về ở trên, mỗi khi có sự thay đổi dữ liệu từ những objects này ORM sẽ gửi những thay đổi đó đến Relational database.

- Hiểu một cách đơn giản thì ORM là công cụ tự động tạo ra các Objects trong ứng dựng từ tables,views, store procedures của một Relational Database nào đó.

2.2 ORM là gì?

- ORM là một công cụ dành cho việc lưu trữ dữ liệu từ domain objects tới CSDL quan hệ như MS SQL theo một cách tự động hóa và không phải lập trình nhiều. ORM bao gồm ba phần chính: những đối tượng Domain class, những đối tượng CSDL quan hệ và thông tin Mapping về cách làm thếnào domain objects nối với những đối tượng CSDL quan hệ (tables, views & storedprocedures). ORM cho phép chúng ta giữ thiết kế CSDL tách biệt với thiết kế domain class. Điều này giúp cho việc bảo trì và nâng cấp ứng dụng. Nó cũng tự động hóa các thao tác CRUD (Create, Read, Update & Delete) giúp cho các lập trình viên không cần phải viết thủ công.

2.3 Cấu trúc của Entity Framework Core



Hình 2.3: Entity Framework Core

- EDM (Entity Data Model): EDM gồm ba phần chính – Conceptual model, Mapping và Storage model.

- Conceptual Model: chứa các model class và những quan hệ của nó. Phần này sẽ độc lập với thiết kế bảng CSDL của bạn.

- Storage Model: là database design model gồm các bảng, views, stored procedures, và những quan hệ của nó và các khóa.

- Mapping: gồm có thông tìn về cách làm thế nào Conceptual model nối với Storage model.

- LINQ to Entities: là một ngôn ngữ truy vấn sử dụng để viết các truy vấn tới object model. Nó trả về các thực thể được định nghĩa trong Conceptual model. Bạn có thể sử dụng kỹ năng LINQ của bạn ở đây.

- Entity SQL: là một ngôn ngữ truy vấn khác giống LINQ to Entities. Tuy nhiên nó có một chút khó khăn hơn L2E và các lập trình viên sẽ phải học nó riêng.

- Object Service: là một điểm vào chính cho việc cho việc truy cập dữ liệu từ CSDL và trả về. Object service có trách nhiệm trong việc cụ thể hóa quá trình chuyển đổi dữ liệu trả về từ một entity client data provider (lớp tiếp theo) tới một entity object structure.

- Entity Client Data Provider: Trách nhiệm chính của lớp này là chuyển đổi L2E hoặc những truy vấn Entity SQL vào một truy vấn SQL , nó được hiểu bởi CSDL cơ bản. Nó giao tiếp với ADO.Net data provider lần lượt gửi và nhận dữ liệu từ CSDL.

- ADO.Net Data Provider: Lớp này giao tiếp với CSDL bằng việc sử dụng chuẩn ADO.Net.

2.4 Mô hình code (Mô hình 3 lớp kết hợp với mô hình MVVM)

- Mô hình MVVM: Kể từ khi Microsoft giới thiệu hai nền tảng phát triển ứng dụng mới là WPF và Silverlight, đã có nhiều thay đổi trong việc xử lý sự kiện và binding dữ liệu, giữa các tầng của ứng dụng với nhau. Qua đó, hầu hết các công việc của tầng kết hợp với lớp presentation. Điều này làm nảy sinh ra nhu cầu phải có một mô hình phát triển ứng dụng mới phù hợp hơn. Và do đó, Model – View – ViewModel (MVVM) pattern ra đời và ngày càng trở nên phổ biến.

- Mô hình 3 lớp: Gồm có 3 thành phần: Presentation Layers, Business Logic Layers, và Data Access Layers.

*+* *Presentation Layers:* Lớp này làm nhiệm vụ giao tiếp với người dùng cuối để thu thập dữ liệu và hiển thị kết quả/dữ liệu thông qua các thành phần trong giao diện người sử dụng.

*+ Busines Logic Layers:* Đây là layer xử lý chính các dữ liệu trước khi được đưa lên hiển thị trên màn hình hoặc xử lý các dữ liệu trước khi chuyển xuống Data Access Layer để lưu dữ liệu xuống cơ sở dữ liệu. Đây cũng là nơi đê kiểm tra ràng buộc, các yêu cầu nghiệp vụ, tính toán, xử lý các yêu cầu và lựa chọn kết quả trả về cho Presentation Layers.

*+ Data Access Layers:* Lớp này thực hiện các nghiệp vụ liên quan đến lưu trữ và truy xuất dữ liệu của ứng dụng như đọc, lưu, cập nhật cơ sở dữ liệu. Cách thức vận hành như sau:

• Đầu tiên User giao tiếp với Presentation Layers (GUI) để gửi đi thông tin và yêu cầu. Tại layer này, các thông tin sẽ được kiểm tra, nếu OK chúng sẽ được chuyển xuống Business Logic Layer (BLL).

• Tại BLL, các thông tin sẽ được nhào nặn, tính toán theo đúng yêu cầu đã gửi, nếu không cần đến Database thì BLL sẽ gửi trả kết quả về GUI, ngược lại nó sẽ đẩy dữ liệu (thông tin đã xử lý) xuống Data Access Layer (DAL).

• DAL sẽ thao tác với Database và trả kết quả về cho BLL, BLL kiểm tra và gửi nó lên GUI để hiển thị cho người dùng.

• Một khi gặp lỗi (các trường hợp không đúng dữ liệu) thì đang ở layer nào thì quăng lên trên layer cao hơn nó 1 bậc cho tới GUI thì sẽ quăng ra cho người dùng biết - Các dữ liệu được trung chuyển giữa các Layer thông qua một đối tượng gọi là Data Transfer Object (DTO), đơn giản đây chỉ là các Class đại diện cho các đối tượng được lưu trữ trong Database.

2.5 Một số package sử dụng trong project.

**•** <PackageReference Include="BootstrapNotifications" Version="0.3.2" />

**•** <PackageReference Include="Microsoft.AspNetCore.Diagnostics.EntityFrameworkCore" Version="3.1.4" />

**•** <PackageReference Include="Microsoft.AspNetCore.Diagnostics.EntityFrameworkCore" Version="3.1.4" />

**•** <PackageReference Include="Microsoft.AspNetCore.Mvc.Razor.RuntimeCompilation" Version="3.1.4" />

**•** <PackageReference Include="Microsoft.AspNetCore.Session" Version="2.2.0" />

**•** <PackageReference Include="Microsoft.EntityFrameworkCore" Version="3.1.4" />

**•** <PackageReference Include="Microsoft.EntityFrameworkCore.Analyzers" Version="3.1.4" />

**•** <PackageReference Include="Microsoft.EntityFrameworkCore.SqlServer" Version="3.1.4" />

**•** <PackageReference Include="Microsoft.EntityFrameworkCore.SqlServer.Design" Version="2.0.0-preview1-final" />

**•** <PackageReference Include="Microsoft.EntityFrameworkCore.Tools" Version="3.1.4">

**•** <PackageReference Include="Microsoft.VisualStudio.Web.BrowserLink" Version="2.2.0" />

**•** <PackageReference Include="Microsoft.VisualStudio.Web.CodeGeneration.Design" Version="3.1.3" />

**•** <PackageReference Include="Microsoft.VisualStudio.Web.CodeGeneration.EntityFrameworkCore" Version="3.1.3" />

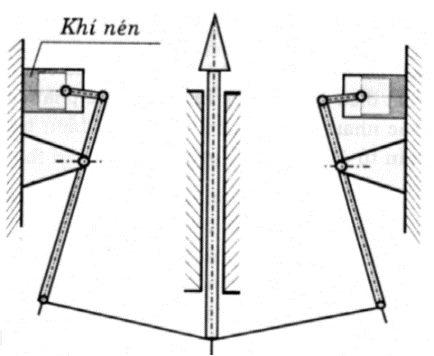
**•** <PackageReference Include="X.PagedList" Version="8.0.7" />

CHƯƠNG 2 (Style “Cap 1\_Chuong”)

Tựa đề chương 2 (Style “Cap 1”)

2.1. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

 Hình ảnh click phải chọn Wrap text -> In line with text, Style “Temp\_Hinh”

Hình 2.1: Nội dung hình tô đen thứ tự, dùng style “Hinh”

2.2. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

2.2.1. Mục lục cấp 3 (Style “Cap 3”)

Nội dung sử dụng (Style “Doan”).

2.2.1.1. Mục lục cấp 4 (Style “Cap 4”)

Nội dung sử dụng (Style “Doan”).

2.2.1.1.1. Mục lục cấp 5 (Style “Cap 5”)

Nội dung sử dụng (Style “Doan”).

Bảng 2.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”.

|  |  |  |
| --- | --- | --- |
| **TT** | **Đặc tính của bảng** | **Ghi chú** |
| 1 | Đặc tính 1 | Before = after = 0, Single |
| 2 | Đặc tính 2 |  |
| 3 | Đặc tính 3 |  |

*\*: Footnote của bảng*

Công thức bình thường dùng Style “Doan”. Còn công thức muốn đánh số thứ tự dùng Style “Cong thuc” đã chỉnh sẵn tab.

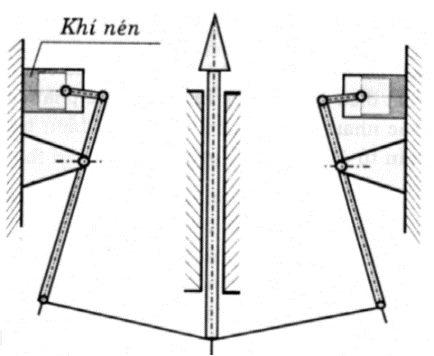
(2.1)

CHƯƠNG 3 (Style “Cap 1\_Chuong”)

Tựa đề chương 3 (Style “Cap 1”)

3.1. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

 Hình ảnh click phải chọn Wrap text -> In line with text, Style “Temp\_Hinh”

Hình 3.1: Nội dung hình tô đen thứ tự, dùng style “Hinh”

3.2. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

3.2.1. Mục lục cấp 3 (Style “Cap 3”)

Nội dung sử dụng (Style “Doan”).

3.2.1.1. Mục lục cấp 4 (Style “Cap 4”)

Nội dung sử dụng (Style “Doan”).

3.2.1.1.1. Mục lục cấp 5 (Style “Cap 5”)

Nội dung sử dụng (Style “Doan”).

Bảng 3.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”.

|  |  |  |
| --- | --- | --- |
| **TT** | **Đặc tính của bảng** | **Ghi chú** |
| 1 | Đặc tính 1 | Before = after = 0, Single |
| 2 | Đặc tính 2 |  |
| 3 | Đặc tính 3 |  |

*\*: Footnote của bảng*

Công thức bình thường dùng Style “Doan”. Còn công thức muốn đánh số thứ tự dùng Style “Cong thuc” đã chỉnh sẵn tab.

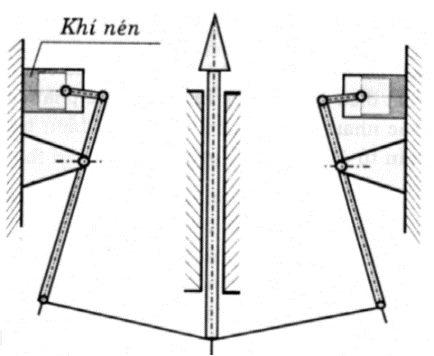
(3.1)

CHƯƠNG 4 (Style “Cap 1\_Chuong”)

Tựa đề chương 4 (Style “Cap 1”)

4.1. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

 Hình ảnh click phải chọn Wrap text -> In line with text, Style “Temp\_Hinh”

Hình 4.1: Nội dung hình tô đen thứ tự, dùng style “Hinh”

4.2. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

4.2.1. Mục lục cấp 3 (Style “Cap 3”)

Nội dung sử dụng (Style “Doan”).

4.2.1.1. Mục lục cấp 4 (Style “Cap 4”)

Nội dung sử dụng (Style “Doan”).

4.2.1.1.1. Mục lục cấp 5 (Style “Cap 5”)

Nội dung sử dụng (Style “Doan”).

Bảng 4.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”.

|  |  |  |
| --- | --- | --- |
| **TT** | **Đặc tính của bảng** | **Ghi chú** |
| 1 | Đặc tính 1 | Before = after = 0, Single |
| 2 | Đặc tính 2 |  |
| 3 | Đặc tính 3 |  |

*\*: Footnote của bảng*

Công thức bình thường dùng Style “Doan”. Còn công thức muốn đánh số thứ tự dùng Style “Cong thuc” đã chỉnh sẵn tab.

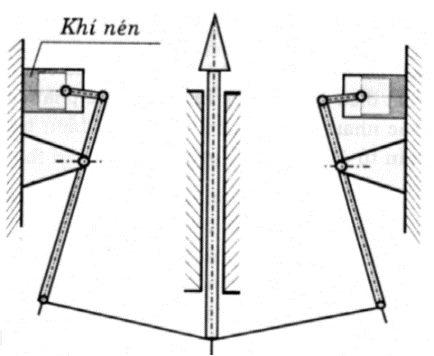
(4.1)

CHƯƠNG 5 (Style “Cap 1\_Chuong”)

Tựa đề chương 2 (Style “Cap 1”)

5.1. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

 Hình ảnh click phải chọn Wrap text -> In line with text, Style “Temp\_Hinh”

Hình 5.1: Nội dung hình tô đen thứ tự, dùng style “Hinh”

5.2. Mục lục cấp 2 (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

5.2.1. Mục lục cấp 3 (Style “Cap 3”)

Nội dung sử dụng (Style “Doan”).

5.2.1.1. Mục lục cấp 4 (Style “Cap 4”)

Nội dung sử dụng (Style “Doan”).

5.2.1.1.1. Mục lục cấp 5 (Style “Cap 5”)

Nội dung sử dụng (Style “Doan”).

Bảng 5.1: Nội dung bảng tô đen thứ tự, dùng style “Bang”.

|  |  |  |
| --- | --- | --- |
| **TT** | **Đặc tính của bảng** | **Ghi chú** |
| 1 | Đặc tính 1 | Before = after = 0, Single |
| 2 | Đặc tính 2 |  |
| 3 | Đặc tính 3 |  |

*\*: Footnote của bảng*

Công thức bình thường dùng Style “Doan”. Còn công thức muốn đánh số thứ tự dùng Style “Cong thuc” đã chỉnh sẵn tab.

(5.1)

CHƯƠNG 6 (Style “Cap 1\_Chuong”)

KẾT LUẬN VÀ KIẾN NGHỊ (Style “Cap 1”)

6.1. Kết luận (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

6.2. Kiến nghị (Style “Cap 2”)

Nội dung sử dụng (Style “Doan”).

TÀI LIỆU THAM KHẢO (Style “Cap 1”)

1. Style “TLTK”
2. Style “TLTK”