

KHOA KỸ THUẬT VÀ CÔNG NGHỆ
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH
HỌC KỲ 2, NĂM HỌC 2024-2025

**Tìm hiểu ngôn ngữ Markdown và
công cụ Markmap để xây dựng sơ đồ
tư duy cho môn học Thiết kế web**

Giảng viên hướng dẫn:
ThS. Phạm Thị Trúc Mai

Sinh viên thực hiện:
Họ tên: Nguyễn Quốc Dũng
MSSV: 170123424
Lớp: DX23TT10

Trà Vinh, tháng 4 năm 2025

KHOA KỸ THUẬT VÀ CÔNG NGHỆ
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH
HỌC KỲ 2, NĂM HỌC 2024-2025

**Tìm hiểu ngôn ngữ Markdown và
công cụ Markmap để xây dựng sơ đồ
tư duy cho môn học Thiết kế web**

Giảng viên hướng dẫn:
ThS. Phạm Thị Trúc Mai

Sinh viên thực hiện:
Họ tên: Nguyễn Quốc Dũng
MSSV: 170123424
Lớp: DX23TT10

Trà Vinh, tháng 4 năm 2026

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

[illegible]

Trà Vinh, ngày tháng năm

Giáo viên hướng dẫn
(Ký tên và ghi rõ họ tên)

NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

[illegible]

Trà Vinh, ngày tháng năm
Thành viên hội đồng
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Trước hết, em xin chân thành cảm ơn các thầy, cô giáo khoa Công Nghệ Thông Tin trường Đại học Trà Vinh đã tạo điều kiện thuận lợi để em có thể hoàn thành báo cáo này.

Em xin gửi lời cảm ơn sâu sắc đến cô hướng dẫn Phạm Thị Trúc Mai vì đã tận tình hướng dẫn, góp ý và hỗ trợ em trong suốt quá trình thực hiện báo cáo. Những kiến thức và kinh nghiệm quý báu từ cô là nền tảng giúp em hoàn thành tốt nội dung bài làm.

Dù đã cố gắng hết sức, nhưng do hạn chế về thời gian và kiến thức, báo cáo không tránh khỏi những thiếu sót. Rất mong nhận được sự cảm thông cũng như sự đóng góp, chỉ dẫn của quý Thầy Cô và các bạn.

Xin chân thành cảm ơn!

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN.....	10
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT.....	11
2.1 Ngôn ngữ đánh dấu Markdown.....	11
2.1.1 Khái niệm.....	11
2.1.2 Một số đặc điểm nổi bật.....	11
2.1.3 Một số cú pháp Markdown phổ biến	12
2.2 Công cụ Markmap	13
2.2.1 Khái niệm.....	13
2.2.2 Nguyên lý hoạt động.....	13
2.2.3 Ưu điểm.....	13
2.2.4 Cài đặt và sử dụng Markmap trong Visual Studio Code (VSCode)...	14
2.3 Sơ đồ tư duy (Mindmap) trong học tập.....	17
2.3.1 Khái niệm sơ đồ tư duy.....	17
2.3.2 Vai trò của sơ đồ tư duy trong học tập.....	17
2.3.3 Cách sử dụng sơ đồ tư duy để học tập hiệu quả:	18
2.3.4 Ứng dụng trong học tập Thiết kế web:	18
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	20
3.1. Xác định những yêu cầu của đề tài:	20
3.2. Quy trình thực hiện:	20
3.2.1. Cài đặt môi trường làm việc:	20
3.2.2. Soạn thảo nội dung sơ đồ tư duy bằng Markdown:	21
3.2.3. Sử dụng công cụ markmap:	24
3.2.4. Chuyển file markdown sang định dạng html:	25
3.2.5. Đưa dự án lên github và tiến hành chạy dự án trực tiếp trên git:	26
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU.....	34
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	36
DANH MỤC TÀI LIỆU THAM KHẢO	38
PHỤ LỤC.....	39

DANH MỤC HÌNH ẢNH

Hình 3. 1: Nhánh thứ nhất của markmap thiết kế web	22
Hình 3. 2: Nhánh thứ hai của markmap thiết kế web	22
Hình 3. 3: Nhánh thiết kế web thủ công	23
Hình 3. 4: Nhánh thứ nhất của markmap thiết kế web	24
Hình 3. 5: Nhánh cuối lưu ý trong thiết web markmap	24
Hình 3. 6: Sơ đồ markmap sau khi hoàn thành.	25
Hình 3. 7: Thực tế xuất file html từ file md có sẵn.....	26
Hình 3. 8: Github sau khi hoàn tất build môi trường chạy html	32
Hình 3. 9: Kết quả chạy file html markmap trên môi trường của Github.....	33

DANH MỤC BẢNG BIỂU

Bảng 2. 1: Cú pháp cơ bản của Markdown.....	12
Bảng 3. 1: Câu lệnh git init	29
Bảng 3. 2: Câu lệnh git add.....	29
Bảng 3. 3: Câu lệnh git init	29
Bảng 3. 4: Cú pháp git remote.	30
Bảng 3. 5: Cú pháp git push.....	30
Bảng 3. 6: URL trang web chạy file html trên github.....	32
Bảng 3. 7: URL khi không phải file index trên github	33

TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH

Đề tài "Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web" tập trung nghiên cứu cách sử dụng ngôn ngữ đánh dấu Markdown kết hợp với công cụ Markmap nhằm trực quan hóa kiến thức môn học dưới dạng sơ đồ tư duy (Mindmap).

Nội dung thực hiện bao gồm:

- Tìm hiểu về ngôn ngữ Markdown và cách công cụ Markmap hoạt động.
- Cài đặt Visual Studio Code và các extension hỗ trợ Markdown và Markmap.
- Sử dụng kiến thức môn Thiết kế web để tạo dữ liệu và biểu diễn nó dưới dạng sơ đồ tư duy bằng Markmap.
- Nộp lại toàn bộ mã nguồn và các tài liệu liên quan theo yêu cầu của khóa học.

MỞ ĐẦU

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc ghi chép, sắp xếp và trình bày thông tin một cách trực quan, dễ hiểu ngày càng trở nên cần thiết, đặc biệt là đối với ngành giáo dục và công nghệ thông tin. Markdown là một ngôn ngữ đánh dấu đơn giản, dễ dàng hiểu, cho phép người học có thể tạo ra các nội dung văn bản rõ ràng, gọn gàng. Kết hợp với công cụ Markmap – một tiện ích giúp chuyển đổi nội dung Markdown sang bản đồ tư duy (Mindmap) trực quan – người học có thể nâng cao kỹ năng tư duy và hệ thống hoá kiến thức hiệu quả hơn.

Bên cạnh đó, ngôn ngữ Markdown – một ngôn ngữ đánh dấu đơn giản, dễ hiểu, dễ viết – cũng được ứng dụng phổ biến trong việc ghi chú, viết văn bản, và trình bày nội dung kỹ thuật. Markdown giúp người sử dụng tạo ra văn bản có định dạng rõ ràng chỉ với cú pháp đơn giản, không yêu cầu nhiều kiến thức lập trình. Đặc biệt, khi sử dụng Markdown với công cụ Markmap – một thư viện mã nguồn mở cho phép chuyển đổi tài liệu Markdown sang sơ đồ tư duy – người dùng có thể chuyển hoá mọi ghi chú lý thuyết sang sơ đồ tư duy nhanh chóng và dễ dàng.

Với mục tiêu áp dụng công nghệ vào việc học tập và củng cố kiến thức môn Thiết kế web, đề tài "Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web" được lựa chọn và thực hiện.

Đề tài không chỉ giúp sinh viên hệ thống lại kiến thức môn học mà còn phát triển tư duy ứng dụng, tăng cường năng lực tự nghiên cứu, và nâng cao khả năng trình bày khoa học. Đồng thời, việc làm quen với các công cụ trực quan như Markdown và Markmap cũng là điều cần thiết giúp sinh viên khi tham gia vào môi trường làm việc thực tế, vì vậy việc ghi chép, truyền đạt và trình bày dữ liệu kỹ thuật một cách rõ ràng là rất quan trọng.

Ngoài phần mở đầu trình bày lí do chọn đề tài, mục đích, đối tượng và phạm vi nghiên cứu và danh mục tài liệu tham khảo bao gồm các tài liệu được trích dẫn, sử dụng và đề cập đến để trao đổi trong đề tài, nội dung báo cáo được chia thành 5 chương:

CHƯƠNG 1: TỔNG QUAN: Giới thiệu khái quát về vấn đề sẽ tập trung nghiên cứu và giải quyết.

CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT: Trình bày cơ sở lý thuyết, lý luận, giả thiết khoa học, các công cụ, kỹ thuật, phần mềm được sử dụng trong Đồ án.

CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU: Mô tả các bước nghiên cứu đã thực hiện, các bản vẽ thiết kế, sơ đồ cài đặt phần mềm hoặc sản phẩm nghiên cứu. Đối với các đề tài nghiên cứu có kết quả là sản phẩm phần mềm cần có bản vẽ thiết kế, cài đặt, ... theo các loại sơ đồ, mô hình thông dụng trong ngành.

CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU: Trình bày các kết quả thu được từ việc thực hiện đề án. Có thể đánh giá về hiệu suất, trải nghiệm người dùng, hoặc trình bày các giao diện chức năng của đề tài ở phần này.

CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN: Trình bày những thành quả đạt được, những thành tựu mới và những kiến nghị mới. Phần tổng kết cần ngắn gọn, không có lời bình và thảo luận bổ sung. Phần hướng nghiên cứu ghi những khuyến nghị cho những hướng nghiên cứu kế tiếp.

Rất mong nhận được các ý kiến đóng góp xây dựng của quý thầy cô trong trường và của các bạn để chương trình được ngày một tốt hơn.

Em xin chân thành cảm ơn.

CHƯƠNG 1: TỔNG QUAN

Trong thời đại công nghệ thông tin ngày càng phát triển, việc hệ thống hoá và phân tích thông tin trở nên cực kỳ cần thiết, nhất là trong lĩnh vực thiết kế web với khối lượng kiến thức đồ sộ và phức tạp. Đồ án này tập trung nghiên cứu và ứng dụng ngôn ngữ đánh dấu Markdown, một cú pháp văn bản đơn giản và mạnh mẽ, kết hợp với công cụ Markmap, một thư viện JavaScript giúp chuyển đổi các file Markdown sang sơ đồ tư duy một cách nhanh chóng và dễ dàng.

Đề tài “Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web” được thực hiện nhằm mục tiêu ứng dụng công nghệ vào việc học tập, đồng thời rèn luyện kỹ năng sử dụng công cụ hỗ trợ lập trình hiện đại như Visual Studio Code và các phần mở rộng liên quan.

Thông qua đề tài môn học, sinh viên sẽ:

- Tìm hiểu cú pháp, cách dùng ngôn ngữ Markdown và công cụ Markmap;
- Cài đặt và cấu hình môi trường làm việc với Visual Studio Code;
- Vận dụng kiến thức đã học từ môn Thiết kế web để xây dựng cơ sở dữ liệu demo cho sơ đồ tư duy
- Hoàn thiện mã nguồn và nội dung báo cáo theo yêu cầu của học phần đồ án cơ sở ngành.

Đây không những là dịp sinh viên trau dồi kiến thức chuyên ngành, mà còn giúp sinh viên rèn luyện khả năng tự học, tự tìm tòi, sáng tạo trong việc tiếp thu và truyền đạt kiến thức.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1 Ngôn ngữ đánh dấu Markdown

2.1.1 Khái niệm

Markdown là một ngôn ngữ đánh dấu (markup language) đơn giản, dễ hiểu và dễ sử dụng, được phát triển bởi John Gruber và Aaron Swartz vào năm 2004. Mục tiêu của Markdown là giúp người viết dễ dàng tạo các nội dung có định dạng (formatting) mà không phải dùng đến thẻ HTML phức tạp. Nội dung viết bởi Markdown có thể dễ dàng chuyển đổi sang HTML, PDF hoặc các định dạng khác.

2.1.2 Một số đặc điểm nổi bật

Một số lý do khiến cho mọi người lựa chọn và sử dụng ngôn ngữ lập trình Markdown:

- Cú pháp đơn giản, dễ học
- Dễ đọc ngay cả ở dạng thuần văn bản
- Hỗ trợ đa nền tảng và tích hợp trong nhiều công cụ (GitHub, Notion, VS Code, Obsidian, v.v.)
- Dễ dàng tích hợp với các công cụ mở rộng như Markmap, Pandoc, hoặc Markdown Preview.

2.1.3 Một số cú pháp Markdown phổ biến

Chức năng	Cú pháp
Tiêu đề	#, ##, ###...
In đậm	**chữ đậm** hoặc __chữ đậm__
In nghiêng	<i>*chữ nghiêng*</i> hoặc <i>_chữ nghiêng_</i>
Danh sách không thứ tự	- mục, + mục, * mục
Danh sách có thứ tự	1. mục, 2. mục
Liên kết	[text](URL)
Hình ảnh	![alt text](image URL)
Trích dẫn	>
Mã nguồn	`code`
Đường kẻ ngang	---, ***, ____
Bảng biểu	Header 1 Header 2 \n --- ---

Bảng 2. 1: Cú pháp cơ bản của Markdown

2.2 Công cụ Markmap

2.2.1 Khái niệm

Markmap là một công cụ mã nguồn mở dùng để trực quan hóa nội dung Markdown dưới dạng sơ đồ tư duy (Mindmap). Khi kết hợp Markmap với nội dung viết bằng Markdown, người dùng có thể tự động tạo ra sơ đồ động, có thể tương tác (click để mở rộng/thu gọn nhánh), dễ dàng trình bày và học tập.

2.2.2 Nguyên lý hoạt động

Markmap hoạt động bằng cách phân tích cấu trúc tiêu đề (#, ##, ###,...) và danh sách trong tập tin Markdown để dựng nên cây sơ đồ tư duy. Mỗi nhánh (node) trong sơ đồ đại diện cho một ý chính hoặc ý phụ.

2.2.3 Ưu điểm

Markmap có một vài ưu điểm nổi bật mà có thể bạn sẽ thấy hữu ích đây:

- **Trực quan và dễ nhìn:** Markmap biến markdown của bạn thành sơ đồ tư duy một cách nhanh chóng. Nhờ đó, bạn có thể dễ dàng hình dung cấu trúc thông tin, mối liên hệ giữa các ý tưởng và nắm bắt nội dung tổng quan một cách trực quan.
- **Tập trung vào nội dung:** Vì dựa trên cú pháp markdown đơn giản, bạn có thể tập trung hoàn toàn vào việc viết và sắp xếp ý tưởng mà không cần lo lắng về các công cụ vẽ phức tạp.
- **Dễ dàng chỉnh sửa và cập nhật:** Khi bạn thay đổi nội dung markdown, sơ đồ tư duy cũng sẽ tự động cập nhật theo. Điều này giúp bạn dễ dàng duy trì và phát triển sơ đồ tư duy của mình theo thời gian.
- **Tính linh hoạt cao:** Bạn có thể sử dụng Markmap cho nhiều mục đích khác nhau, từ ghi chú nhanh, tóm tắt sách, lên kế hoạch dự án đến phác thảo bài thuyết trình.
- **Khả năng nhúng và chia sẻ:** Sơ đồ tư duy Markmap có thể dễ dàng nhúng vào các trang web hoặc chia sẻ dưới dạng tệp HTML.

- **Miễn phí và mã nguồn mở:** Markmap là một công cụ miễn phí và mã nguồn mở, bạn có thể sử dụng và tùy chỉnh nó theo nhu cầu của mình.

2.2.4 Cài đặt và sử dụng Markmap trong Visual Studio Code (VSCode)

Cài đặt Visual Studio Code:

Bước 1: Truy cập trang tải xuống: Mở trình duyệt web (ví dụ: Chrome, Firefox, Edge) và truy cập trang web chính thức của Visual Studio Code: <https://code.visualstudio.com/>

Bước 2: Tải xuống phiên bản phù hợp với máy: Trang web sẽ tự động nhận diện hệ điều hành của bạn. Nhấp vào nút màu xanh lam lớn có chữ **"Download for Windows"**. Nếu không, bạn có thể nhấp vào mũi tên xuống bên cạnh nút và chọn phiên bản phù hợp (User Installer, System Installer, hoặc .zip).

Bước 3: Chạy trình cài đặt: Sau khi tải xuống hoàn tất, hãy nhấp đúp vào tệp .exe vừa tải xuống để chạy trình cài đặt.

Bước 4: Đọc và chấp nhận thỏa thuận: Đọc kỹ thỏa thuận cấp phép và đánh dấu vào ô **"I accept the agreement"** (Tôi chấp nhận thỏa thuận). Sau đó, nhấp vào **"Next"**.

Bước 5: Chọn thư mục cài đặt: Có thể giữ nguyên thư mục cài đặt mặc định hoặc nhấp vào **"Browse..."** để chọn một thư mục khác. Nhấp vào **"Next"**.

Bước 6: Chọn các tác vụ bổ sung: Màn hình lúc này sẽ hiển thị cho phép tùy chọn các tác vụ bổ sung. Các tùy chọn thường được khuyến nghị bao gồm: Create a desktop icon: Tạo biểu tượng trên màn hình desktop để khởi chạy VS Code nhanh chóng, Add "Open with Code" action to Windows Explorer file context menu: Thêm tùy chọn "Open with Code" khi bạn nhấp chuột phải vào một tệp trong File Explorer, Add "Open with Code" action to Windows Explorer directory context menu: Thêm tùy chọn "Open with Code" khi bạn nhấp chuột phải vào một thư mục trong File Explorer, Register Code as an editor for supported file types: Đặt VS Code làm trình soạn thảo mặc định cho các loại tệp được hỗ trợ. Đánh dấu vào các ô bạn muốn và nhấp vào **"Next"**.

Bước 7: Sẵn sàng cài đặt: Nhấp vào nút "Install" để bắt đầu quá trình cài đặt.

Bước 8: Hoàn tất cài đặt: Sau khi quá trình cài đặt hoàn tất, bạn có thể đánh dấu vào ô "Launch Visual Studio Code" nếu bạn muốn mở VS Code ngay lập tức. Nhấp vào "Finish".

Cài đặt các plugin “**Markdown Preview Enhanced**” và **Markmap** hỗ trợ dự án:

Bước 1: Mở VS Code: Khởi động ứng dụng Visual Studio Code trên máy tính.

Bước 2: Truy cập mục Extensions: Nhấp vào biểu tượng hình vuông ở thanh bên trái (hoặc nhấn Ctrl+Shift+X trên Windows/Linux hoặc Cmd+Shift+X trên macOS). Đây là nơi bạn quản lý các extension của VS Code.

Bước 3: Tìm kiếm extension: Trong thanh tìm kiếm, gõ “**Markdown Preview Enhanced**” và “**Markmap**”.

Bước 4: Chọn Markdown Preview Enhanced và Markmap: Chọn extension “**Markdown Preview Enhanced**” được phát triển bởi shd101wyy và extension “**Markmap**” được phát triển bởi Gerald Liu. Hãy nhấp vào 2 extension này.

Bước 5: Cài đặt extension: Trên trang chi tiết của 2 extension, sẽ thấy nút "Install" màu xanh lá cây. Nhấp vào nút này để bắt đầu quá trình cài đặt.

Bước 6: Khởi động lại VS Code (nếu cần): Sau khi cài đặt xong, VS Code có thể yêu cầu bạn khởi động lại để kích hoạt extension. Nếu có thông báo, hãy nhấp vào nút “**Reload**” hoặc đóng và mở lại VS Code.

Cách sử dụng sau khi cài đặt:

- Tạo file markdown cơ bản với đuôi file (.md).
- Nhấp đôi chuột vào file mới tạo để xem phần nội dung cũng như chỉnh sửa file .md.
- Chọn "**Open Preview to the Side**" (hoặc sử dụng phím tắt Ctrl+K V) để xem file .md bằng Markdown Preview Enhanced
- Để có thể sử dụng **Markmap: Open Preview** xem sơ đồ tư duy trực tiếp, có 3 cách:

Cách 1: Command Palette: Nhấn **Ctrl+Shift+P** để mở Command Palette. Sau đó gõ "**Open as markmap**" trong cửa sổ Command Palette và chọn lệnh tương ứng.

Cách 2: Context Menu: Chuột phải vào file Markdown đang mở trong editor hoặc trong Explorer ở thanh bên trái, sau đó chọn "**Open as markmap**".

Cách 3: Button trên Title Bar: Khi đã mở một file Markdown, có thể thấy biểu tượng Markmap trên thanh tiêu đề góc trên bên phải của editor. Nhấn vào biểu tượng đó để mở chế độ xem Markmap.

Lưu ý: Khi bạn chỉnh sửa file Markdown, chế độ xem Markmap sẽ tự động cập nhật.

2.3 Sơ đồ tư duy (Mindmap) trong học tập

2.3.1 Khái niệm sơ đồ tư duy

Sơ đồ tư duy là một kỹ thuật ghi chép và sắp xếp thông tin bằng việc kết hợp hình ảnh, từ ngữ, âm thanh và các nhánh tỏa ra từ một ý tưởng trung tâm.

Sơ đồ tư duy mô phỏng cách bộ não chúng ta liên kết các ý tưởng, giúp bạn nắm bắt thông tin một cách tổng quan và chi tiết.

2.3.2 Vai trò của sơ đồ tư duy trong học tập

Tóm tắt và hệ thống hóa kiến thức: Sơ đồ tư duy giúp bạn cô đọng những thông tin phức tạp thành một cấu trúc dễ nhìn, làm nổi bật mối liên hệ giữa các khái niệm.

Tăng cường khả năng ghi nhớ: Việc sử dụng hình ảnh, màu sắc và từ khóa kích thích cả hai bán cầu não, giúp bạn ghi nhớ thông tin hiệu quả và lâu dài hơn so với việc ghi chép tuyến tính truyền thống.

Phát triển tư duy sáng tạo: Sơ đồ tư duy khuyến khích bạn suy nghĩ đa chiều, tạo ra các liên kết mới giữa các ý tưởng, từ đó thúc đẩy sự sáng tạo và khả năng giải quyết vấn đề.

Cải thiện sự tập trung: Khi vẽ sơ đồ tư duy, bạn tập trung vào việc kết nối các ý tưởng, giúp duy trì sự chú ý và giảm xao nhãng trong quá trình học tập.

Ôn tập hiệu quả: Sơ đồ tư duy là một công cụ tuyệt vời để ôn lại kiến thức một cách nhanh chóng và hiệu quả, giúp bạn dễ dàng nhớ lại các điểm chính và mối liên hệ giữa chúng.

Lập kế hoạch và tổ chức: Bạn có thể sử dụng sơ đồ tư duy để lên kế hoạch học tập, sắp xếp các nhiệm vụ và quản lý thời gian hiệu quả hơn.

Thúc đẩy sự tham gia và hợp tác: Trong học nhóm, sơ đồ tư duy có thể là một công cụ hữu ích để chia sẻ ý tưởng, thảo luận và cùng nhau xây dựng kiến thức.

2.3.3 Cách sử dụng sơ đồ tư duy để học tập hiệu quả:

Xác định chủ đề trung tâm: Bắt đầu bằng cách viết hoặc vẽ chủ đề chính của bài học hoặc nội dung bạn muốn ghi nhớ ở trung tâm trang giấy.

Thêm các nhánh chính: Từ chủ đề trung tâm, vẽ các nhánh lớn tỏa ra, mỗi nhánh đại diện cho một ý chính hoặc một khía cạnh quan trọng của chủ đề. Sử dụng từ khóa ngắn gọn cho mỗi nhánh.

Phát triển các nhánh phụ: Từ các nhánh chính, tiếp tục vẽ các nhánh nhỏ hơn để ghi chú các ý phụ, chi tiết hoặc ví dụ liên quan.

Sử dụng từ khóa và hình ảnh: Ưu tiên sử dụng từ khóa ngắn gọn thay vì cả câu dài. Thêm hình ảnh, biểu tượng và màu sắc để làm cho sơ đồ tư duy trở nên trực quan và dễ nhớ hơn.

Tạo liên kết: Sử dụng các đường kẻ, mũi tên hoặc màu sắc để thể hiện mối liên hệ giữa các ý tưởng khác nhau trong sơ đồ.

Linh hoạt và sáng tạo: Không có quy tắc cứng nhắc nào cho việc vẽ sơ đồ tư duy. Hãy tự do sáng tạo và phát triển sơ đồ theo cách bạn cảm thấy dễ hiểu và phù hợp nhất với phong cách học tập của mình.

Ôn tập thường xuyên: Xem lại sơ đồ tư duy của bạn định kỳ để củng cố kiến thức và ghi nhớ thông tin lâu hơn.

2.3.4 Ứng dụng trong học tập Thiết kế web:

Thiết kế web là môn học chứa nhiều khái niệm liên quan đến như html, css, phải tìm hiểu về bố cục, giao diện người dùng,... Việc áp dụng sơ đồ tư duy vào học thiết kế web có thể giúp:

- **Hệ thống hóa kiến thức phức tạp:** Giúp bạn sắp xếp các khái niệm, nguyên tắc và công nghệ liên quan đến thiết kế web một cách rõ ràng và có cấu trúc.
- **Tăng cường khả năng ghi nhớ:** Sử dụng hình ảnh, màu sắc và từ khóa kích thích não bộ, giúp bạn ghi nhớ thông tin hiệu quả hơn.
- **Khơi gợi ý tưởng sáng tạo:** Tạo không gian trực quan để liên kết các ý tưởng, khám phá các phong cách thiết kế và giải pháp độc đáo.

- **Lập kế hoạch dự án hiệu quả:** Hỗ trợ việc xác định yêu cầu, xây dựng cấu trúc trang, phác thảo wireframe và lên kế hoạch nội dung một cách trực quan.
- **Nắm bắt các nguyên tắc UI/UX:** Giúp bạn hiểu rõ và áp dụng các nguyên tắc về trải nghiệm người dùng và thiết kế giao diện.
- **Ôn tập và xem lại dễ dàng:** Cung cấp một cái nhìn tổng quan về toàn bộ kiến thức, giúp bạn ôn tập nhanh chóng và hiệu quả.
- **Tạo sự kết nối giữa các lĩnh vực:** Giúp bạn liên kết các khía cạnh khác nhau của thiết kế web, từ lý thuyết đến thực hành và công nghệ.

Tóm lại, sơ đồ tư duy là một công cụ mạnh mẽ giúp bạn học thiết kế web một cách trực quan, hiệu quả và sáng tạo hơn.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

Dựa trên nền tảng lý thuyết đã tìm hiểu, đề tài được triển khai thông qua các bước cụ thể để tạo ra sản phẩm sơ đồ tư duy sử dụng ngôn ngữ Markdown và công cụ Markmap. Các bước hiện thực hóa được trình bày như sau:

3.1. Xác định những yêu cầu của đề tài:

Đề tài tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web bao gồm 4 yêu cầu:

- Tìm hiểu ngôn ngữ đánh dấu Markdown và công cụ Markmap.
- Cài đặt Visual Studio Code và các extensions cần thiết để sử dụng ngôn ngữ Markdown và công cụ Markmap.
- Dùng kiến thức đã học của môn Thiết kế web để làm dữ liệu demo cho sơ đồ tư duy (Mindmap).
- Nộp mã nguồn và các nội dung liên quan đến Thực tập đồ án cơ sở ngành theo quy định của khóa học.

3.2. Quy trình thực hiện:

3.2.1. Cài đặt môi trường làm việc:

- Tải và cài đặt VS Code từ trang chính thức: code.visualstudio.com
- Cài đặt các tiện ích mở rộng (extensions):
 - **Markdown Preview Enhanced:** hỗ trợ xem trực tiếp nội dung Markdown.
 - **Markmap VS Code Extension:** hỗ trợ chuyển đổi nội dung Markdown thành sơ đồ tư duy động.
 - **Live Server:** dùng để hiển thị nội dung HTML động.
 - **GitHub Pull Requests and Issues:** giúp quản lý, theo dõi và tương tác với các pull request và issue trên GitHub trực tiếp trong Visual Studio Code.

3.2.2. Soạn thảo nội dung sơ đồ tư duy bằng Markdown:

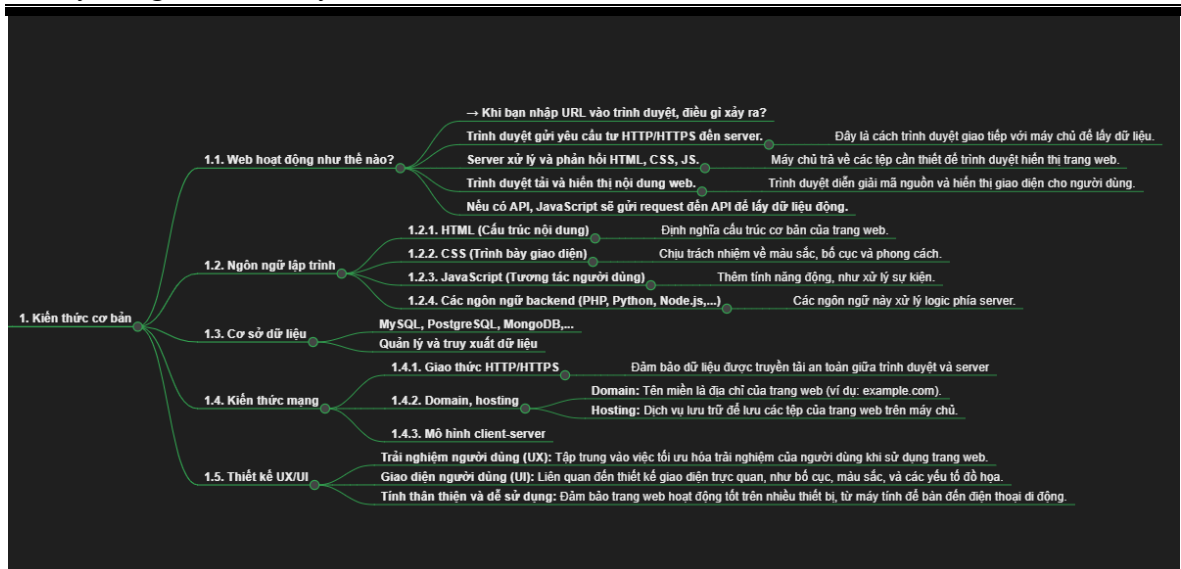
Tiến hành xây dựng nội dung sơ đồ tư duy dựa trên chương trình môn học Thiết kế web, với các chủ đề gồm có:

Phần title của mindmap đề tên **Thiết kế Web** 🖥️

Từ title chia ra thành 5 nhánh chính: 1. Kiến thức cơ bản, 2. Thiết kế web từ công cụ trực tuyến, 3. Thiết kế web thủ công, 4. Thiết kế web dựa trên framework, và nhánh cuối cùng là 📌 Lưu ý.

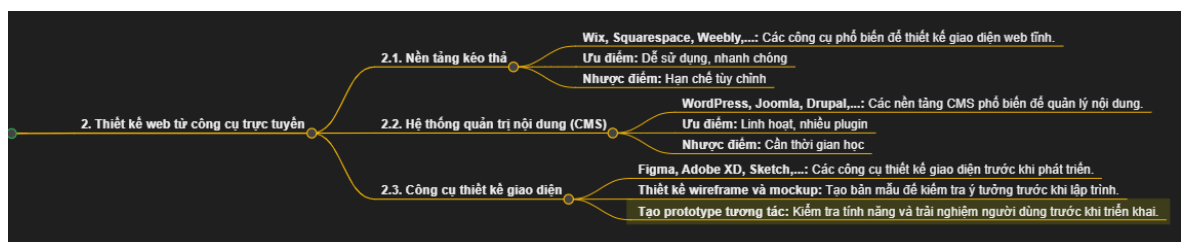
Nhánh chính đầu tiên kiến thức cơ bản gồm 5 nhánh phụ:

- Nhánh phụ thứ nhất là **“Web hoạt động như thế nào?”** mô tả quy trình khi người dùng nhập URL vào trình duyệt, gồm các bước: gửi yêu cầu HTTP, server xử lý và phản hồi, trình duyệt tải và hiển thị nội dung, cũng như việc JavaScript gửi request đến API để lấy dữ liệu động.
- Tiếp theo,, nhánh phụ thứ hai nói về **“Ngôn ngữ lập trình”** giới thiệu các công nghệ chính để phát triển web: HTML cho cấu trúc nội dung, CSS để trình bày giao diện, JavaScript để tạo tương tác người dùng, và các ngôn ngữ backend như PHP, Python, Node.js để xử lý phía máy chủ.
- Nhánh phụ thứ ba là về **“Cơ sở dữ liệu”**. Phần này đề cập đến các hệ quản trị cơ sở dữ liệu phổ biến như MySQL, PostgreSQL, MongoDB, và cách quản lý, truy xuất dữ liệu từ cơ sở dữ liệu.
- Nhánh **“Kiến thức mạng”** trình bày về các yếu tố nền tảng như giao thức HTTP/HTTPS, tên miền (domain), hosting và mô hình hoạt động client-server.
- Cuối cùng là **“Thiết kế UX/UI”** nhấn mạnh tầm quan trọng của việc tạo ra trải nghiệm người dùng (UX) và giao diện người dùng (UI) thân thiện, dễ sử dụng trong thiết kế web hiện đại.



Hình 3. 1: Nhánh thứ nhất của markmap thiết kế web

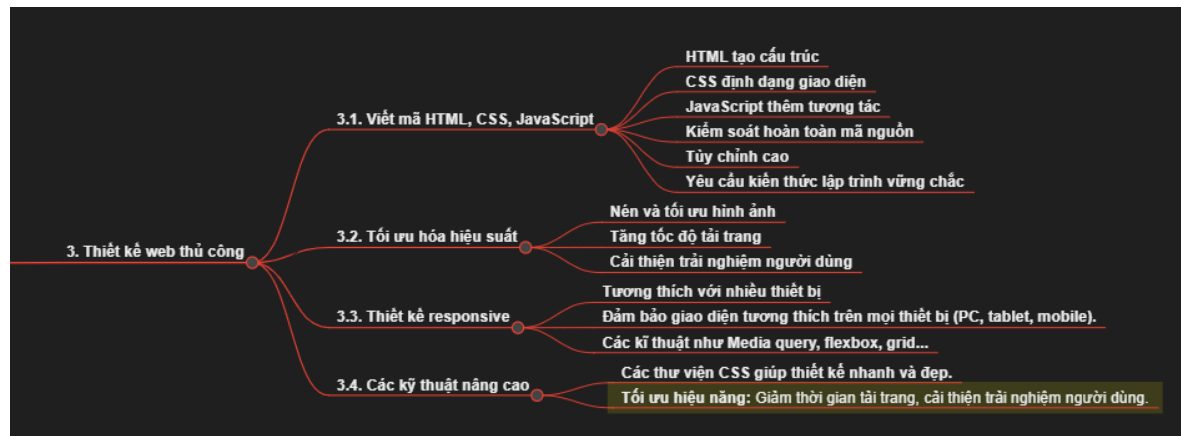
Nhánh chính thứ hai đề cập đến việc **Thiết kế web từ công cụ trực tuyến**. Đây là phương pháp xây dựng website nhanh chóng nhờ vào các nền tảng hoặc công cụ có sẵn. Các **nền tảng kéo thả** như Wix, Squarespace, Weebly,... cho phép người dùng chỉ cần thao tác kéo-thả các thành phần giao diện mà không cần biết lập trình. Ưu điểm nổi bật của phương pháp này là dễ sử dụng và có thể tạo website trong thời gian rất ngắn. Tuy nhiên, nó cũng tồn tại nhược điểm là khả năng tùy chỉnh hạn chế và khó mở rộng nếu yêu cầu phức tạp hơn. Một hướng tiếp cận khác là sử dụng **hệ thống quản trị nội dung (CMS)** như WordPress, Joomla, Drupal,... Đây là các nền tảng linh hoạt, hỗ trợ nhiều plugin giúp mở rộng tính năng website dễ dàng. Tuy nhiên, CMS đòi hỏi người dùng cần có thêm thời gian học tập và làm quen với hệ thống quản trị. Bên cạnh đó, việc thiết kế giao diện còn có thể được hỗ trợ bởi các **công cụ thiết kế giao diện** như Figma, Adobe XD, Sketch,..., giúp tạo ra các bản wireframe, mockup và prototype tương tác để định hình giao diện và trải nghiệm người dùng trước khi triển khai thực tế.



Hình 3. 2: Nhánh thứ hai của markmap thiết kế web

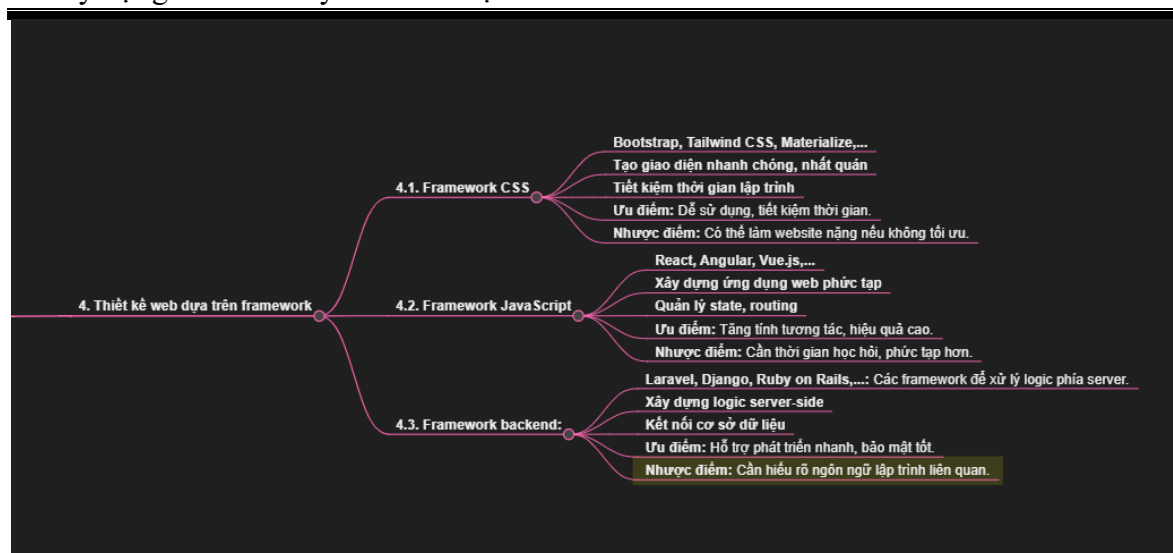
Nhánh chính thứ ba, **Thiết kế web thủ công** là phương pháp xây dựng website bằng cách trực tiếp viết mã HTML, CSS và JavaScript. Ưu điểm của cách

làm này là khả năng kiểm soát toàn bộ mã nguồn, dễ dàng tùy chỉnh giao diện và tính năng theo nhu cầu riêng, đồng thời yêu cầu lập trình viên phải có nền tảng kiến thức lập trình vững chắc. Ngoài ra, thiết kế thủ công cần chú trọng đến **tối ưu hóa hiệu suất** như tối ưu hình ảnh, mã nguồn, tăng tốc độ tải trang và nâng cao trải nghiệm người dùng. Một yếu tố không thể thiếu là **thiết kế responsive** – đảm bảo website hiển thị tốt trên nhiều thiết bị (máy tính, điện thoại, máy tính bảng) bằng cách áp dụng các kỹ thuật như Media Query, Flexbox và Grid.



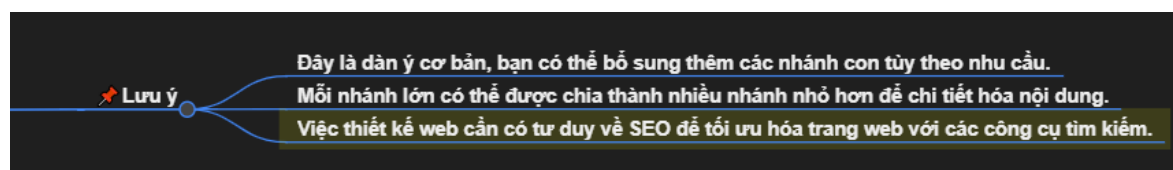
Hình 3. 3: Nhánh thiết kế web thủ công

Nhánh kế tiếp, **Thiết kế web dựa trên framework**. Đây là xu hướng hiện đại nhằm tăng tốc độ phát triển dự án. Các **framework CSS** như Bootstrap, Tailwind CSS, Materialize hỗ trợ tạo giao diện nhanh chóng, chuẩn hóa thiết kế và tiết kiệm thời gian lập trình. Song song đó, **framework JavaScript** như React, Angular, Vue.js cho phép xây dựng các ứng dụng web phức tạp, tối ưu việc quản lý trạng thái và định tuyến trang. Đối với backend, các **framework như Laravel, Django, Ruby on Rails** giúp lập trình viên dễ dàng xây dựng logic server-side và kết nối cơ sở dữ liệu hiệu quả. Dù framework mang lại nhiều **ưu điểm** như tốc độ phát triển nhanh và mã nguồn có tổ chức tốt, nhưng cũng có **nhược điểm** như cần thời gian làm quen và đôi khi bị phụ thuộc vào framework đã chọn.



Hình 3. 4: Nhánh thứ nhất của markmap thiết kế web

Và cuối cùng, phần **Lưu ý**. Đây chỉ là những dàn ý cơ bản, người học hoặc lập trình viên có thể linh hoạt bổ sung thêm các nhánh con tùy theo nhu cầu cụ thể. Ngoài ra, trong thiết kế web hiện đại, tư duy về **SEO** (tối ưu hóa công cụ tìm kiếm) cần được tích hợp ngay từ đầu để website dễ dàng được các công cụ tìm kiếm đánh giá cao và xếp hạng tốt hơn.



Hình 3. 5: Nhánh cuối lưu ý trong thiết web markmap

3.2.3. Sử dụng công cụ markmap:

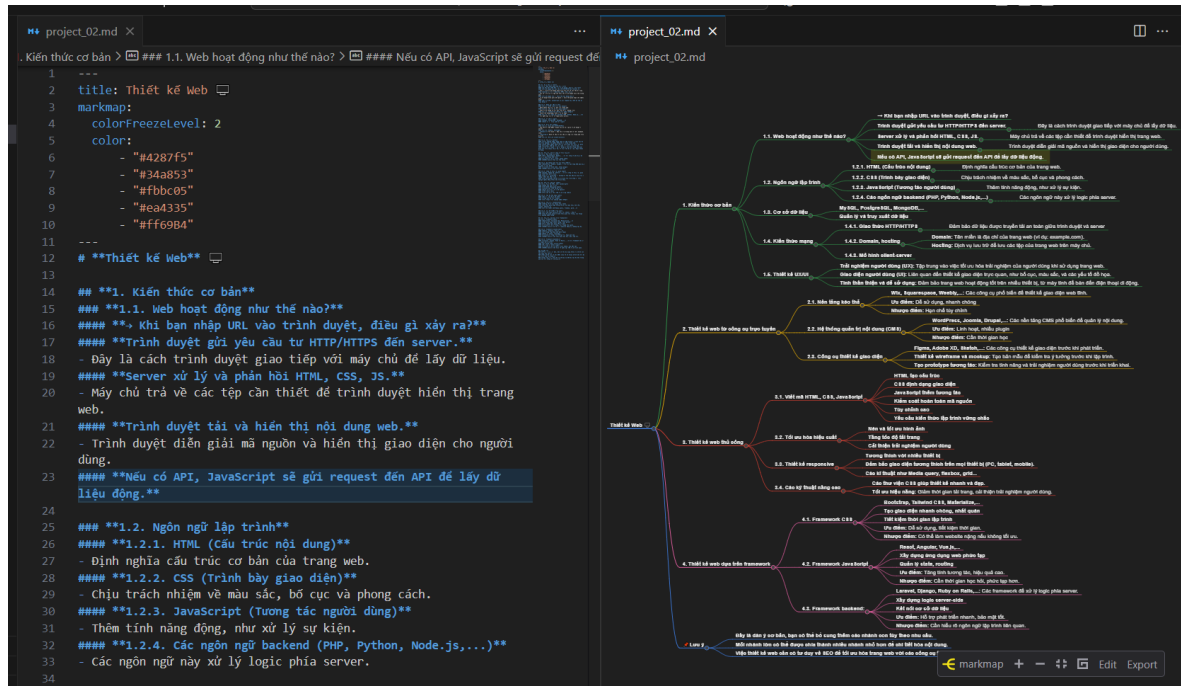
Sau khi đã hoàn thành việc cài đặt môi trường làm việc và soạn thảo nội dung của môn thiết kế web, kế đến sẽ tiến hành thực hiện tạo sơ đồ bằng markmap.

Đầu tiên ta tạo file markdown bằng cách nhân vào biểu tượng New file ở thanh Explorer bên trái của phần mềm Visual Studio Code. Nếu không có thể nhấn tổ hợp phím Ctrl+Shift+E. File được tạo phải có định dạng đuôi .md.

Kế đến điền lần lượt nội dung đề tài mình đã soạn thảo đưa vào file .md đó. Với phần tiêu đề (Headings): Sử dụng dấu # để tạo các nhánh chính và nhánh con. Cấp độ tiêu đề sẽ tương ứng với cấp độ của nhánh trong sơ đồ tư duy: # Tiêu đề chính, ## Nhánh cấp 1, ### Nhánh cấp 2,...

Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web

Sau khi hoàn thành việc đưa nội dung soạn thảo và file markdown, tiếp tục tiến hành kiểm tra file thực hiện đã đúng với mong muốn hay chưa bằng công cụ Markmap. Tìm kiếm biểu tượng Open as markmap ở góc trên bên phải của VSC. Nếu không tìm được, hãy kiểm tra xem extension markmap đã được cài đặt chưa. Nếu Markmap chưa ưng ý hoặc có nội dung cần chỉnh, ta có thể trực tiếp chỉnh sửa trên file .md và Markmap sẽ hiện thị ngay lập tức để giúp chúng ta dễ dàng kiểm tra hơn.



Hình 3. 6: Sơ đồ markmap sau khi hoàn thành.

3.2.4. Chuyển file markdown sang định dạng html:

Sau khi đã chỉnh sửa markmap được ưng ý, kế tiếp chúng ta sẽ tiến hành chuyển đổi file .md trở thành file .html.

Đầu tiên ta mở cửa sổ Terminal bằng cách vào chọn vào thanh công cụ **View** của VSC và chọn Terminal hoặc gõ lệnh **“Ctrl + `”**.

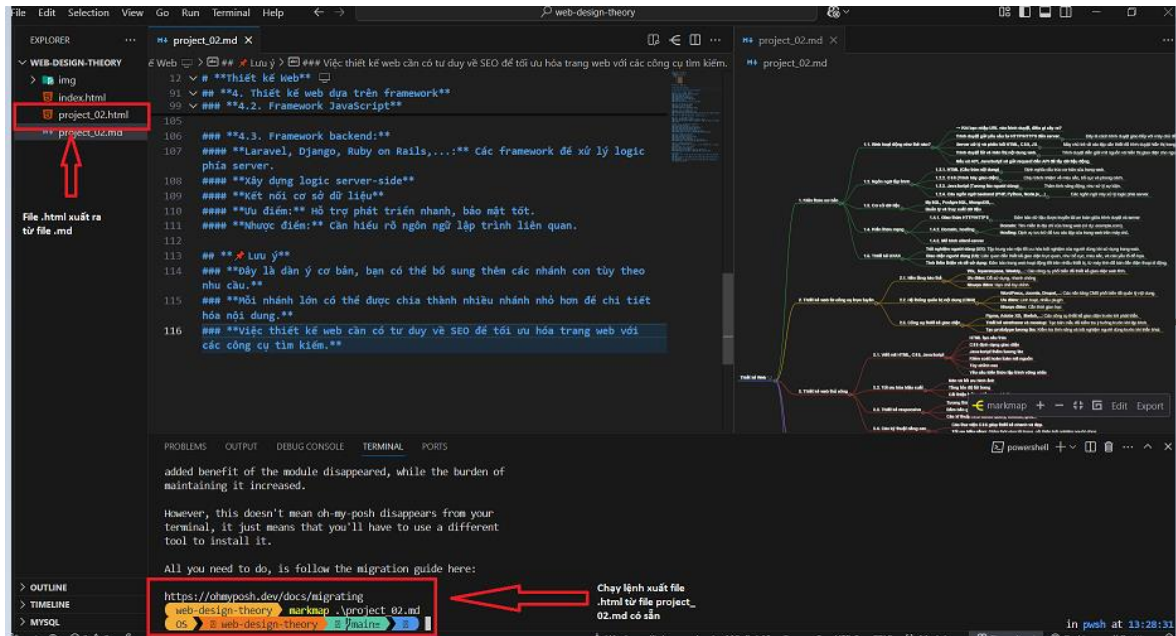
Khi cửa sổ Terminal đã xuất hiện, ta bắt đầu cài đặt tool markmap-cli hỗ trợ việc chuyển đổi file .md sang html. Ta gõ vào terminal câu lệnh:

```
npm install -g markmap-cli
```

Sau khi cài đặt hoàn tất, ta sử dụng câu lệnh:

markmap .\project_02.md

Với .\project_02.md là tên file markmap.



Hình 3. 7: Thực tế xuất file html từ file md có sẵn

Sau khi chạy câu lệnh xong, chúng ta sẽ nhận được file project_02.html và chúng ta có thể mở trực tiếp trên trình duyệt để kiểm tra file có đúng ta mong muốn hay không.

3.2.5. Đưa dự án lên github và tiến hành chạy dự án trực tiếp trên git:

Tạo tài khoản GitHub

- Đầu tiên, Truy cập trang web GitHub: Mở trình duyệt web của bạn và đi đến trang chủ của GitHub: <https://github.com/>
- Nhấp vào nút "Sign up": Ở góc trên cùng bên phải của trang web, bạn sẽ thấy nút "Sign up". Nhấp vào nút này để bắt đầu quá trình tạo tài khoản.
- **Nhập thông tin tài khoản:** Nhập vào những thông tin cá nhân theo yêu cầu trên github:
 - o **Emai:** Nhập địa chỉ email hợp lệ mà bản thân có quyền truy cập. GitHub sẽ sử dụng địa chỉ này để liên lạc với bạn và xác minh tài khoản.

- **Password:** Tạo một mật khẩu mạnh và an toàn. Hãy sử dụng kết hợp chữ hoa, chữ thường, số và ký tự đặc biệt.
- **Username:** Chọn một tên người dùng duy nhất. Tên người dùng này sẽ hiển thị trên trang cá nhân của bạn và trong các hoạt động bạn thực hiện trên GitHub.
- **Your Country/Region:** Lựa chọn quốc gia bản thân đang ở.
- Nhấn chọn vào **Receive occasional product updates and announcements.**
- Tiếp tục nhấn **Continue** và xác minh tài khoản:
 - GitHub sẽ yêu cầu giải một câu đố và thực hiện một bước xác minh khác để đảm bảo bạn không phải là bot. Hãy làm theo hướng dẫn trên màn hình.
 - **Xác minh địa chỉ email:** GitHub sẽ gửi một email đến địa chỉ bạn đã cung cấp để xác minh tài khoản. Mở hộp thư đến của bạn, tìm email từ GitHub và nhấp vào liên kết xác minh trong email đó.
- Sau khi hoàn thành các thao tác trên, bạn sẽ nhận được một giao diện hoàn thành:

Tiến hành tạo Repository trên github và đẩy dự án hoàn thiện lên git:

- **Đăng nhập vào tài khoản GitHub:** Mở trình duyệt web của bạn và truy cập <https://github.com/>. Đăng nhập bằng tên người dùng và mật khẩu bạn đã tạo.
- Ở góc trên cùng bên phải, nhấp vào biểu tượng dấu “+” (dấu cộng) và chọn “**New repository**” từ menu thả xuống.
- **Điền thông tin repository:** Sau khi chuyển đến trang “**Create a new repository**”. Điền các thông tin sau:
 - **Repository name:** Nhập tên bạn muốn đặt cho repository của mình. Tên repo thường ngắn gọn và mô tả mục đích của dự án (ví dụ: my-website, python-scripts,...).

- **Description (tùy chọn):** Bạn có thể thêm một mô tả ngắn gọn về repository của mình. Điều này giúp người khác (và cả bạn sau này) hiểu rõ hơn về dự án.
- **Public hoặc Private: Public:** Bất kỳ ai trên internet đều có thể xem repository của bạn. **Private:** Chỉ những người bạn chỉ định mới có thể xem repository. Chọn tùy chọn phù hợp với nhu cầu của bạn. (Gói miễn phí của GitHub cho phép tạo repository private với số lượng cộng tác viên giới hạn).
- **Initialize this repository with:** Một số tùy chọn trước khi khởi tạo repository:
 - **Add a README file:** Nên chọn tùy chọn này. README là một file thường chứa thông tin giới thiệu về dự án, hướng dẫn cài đặt và sử dụng.
 - **Add .gitignore:** Chọn ngôn ngữ lập trình hoặc framework bạn sẽ sử dụng trong dự án (ví dụ: Python, Node.js, HTML). GitHub sẽ tạo một file .gitignore mẫu để chỉ định các file và thư mục mà Git sẽ bỏ qua (ví dụ: file tạm, thư mục node_modules).
 - **Choose a license:** Chọn một giấy phép mã nguồn mở nếu bạn muốn chia sẻ dự án của mình với người khác. Nếu đây là dự án cá nhân và bạn không chắc chắn, bạn có thể bỏ qua tùy chọn này.
- **Nhấp vào nút “Create repository”:** Sau khi điền đầy đủ thông tin, nhấp vào nút màu xanh lá cây “Create repository” ở cuối trang. Repository mới của bạn sẽ được tạo trên GitHub.

Đẩy File từ VS Code lên GitHub:

Để đẩy file từ VS Code lên repository GitHub vừa tạo, cần đảm bảo rằng thư mục dự án trong VS Code đã được khởi tạo làm một Git repository cục bộ và đã được liên kết với repository GitHub từ xa.

- **Mở thư mục dự án trong VS Code:** Mở VS Code và điều hướng đến thư mục chứa các file dự án bạn muốn đẩy lên GitHub (File > Open Folder...).
- **Khởi tạo Git repository cục bộ (nếu chưa):** Nếu đây là lần đầu tiên bạn làm việc với Git trong dự án này, bạn cần khởi tạo một Git repository cục bộ.
 - o Mở Terminal trong VS Code (Terminal > New Terminal).
 - o Chạy lệnh:

Bash
git init

Bảng 3. 1: Câu lệnh git init

- o Lệnh này sẽ tạo một thư mục .git ẩn trong thư mục dự án của bạn, đánh dấu nó là một Git repository.
- **Thêm các file vào staging area:** Tiếp theo, bạn cần chọn các file bạn muốn commit (lưu lại trạng thái hiện tại) để chuẩn bị cho việc đẩy lên GitHub.
 - o Để thêm tất cả các file chưa được theo dõi:

Bash
git add .

Bảng 3. 2: Câu lệnh git add.

- o **Để thêm một hoặc nhiều file cụ thể:** Thay “.” bằng tên của các file, cách nhau bởi dấu cách (ví dụ: git add index.html style.css script.js).
- **Commit các thay đổi:** Sau khi thêm các file vào staging area, bạn cần commit chúng với một tin nhắn mô tả các thay đổi bạn đã thực hiện.
 - o Chạy lệnh sau:

Bash

```
git commit -m "Initial commit"
```

Bảng 3. 3: Câu lệnh git init

- Thay **“Initial commit”** bằng một tin nhắn mô tả ý nghĩa của commit này (ví dụ: “Add basic HTML structure”, “Implement user authentication”, “Fix bug in login form”).
- **Liên kết repository cục bộ với repository GitHub từ xa:** Bây giờ cần nói cho Git repository cục bộ biết về repository GitHub từ xa vừa tạo.
 - Truy cập trang repository trên GitHub. Sử dụng lệnh:

```
Bash
```

```
git remote add origin
```

```
https://github.com/ YOUR\_USERNAME/ YOUR\_REPOSITORY\_NAME.git
```

Bảng 3. 4: Cú pháp git reomote.

- Thay YOUR_USERNAME bằng tên người dùng GitHub username.
- Thay YOUR_REPOSITORY_NAME bằng tên repository đã tạo.
- **Đẩy các commit lên GitHub:** Cuối cùng, bạn sẽ đẩy các commit từ repository cục bộ của bạn lên repository GitHub từ xa.
 - Chạy lệnh sau:

```
Bash
```

```
git push -u origin main
```

Bảng 3. 5: Cú pháp git push

- git push: Lệnh để đẩy các commit.
- -u origin: Thiết lập origin làm remote mặc định cho nhánh hiện tại.
- main: Tên nhánh mặc định trên GitHub (trước đây thường là master). Nếu repository của bạn sử dụng nhánh master, hãy thay main bằng master.

Sau khi lệnh git push hoàn thành thành công, các file và commit của bạn sẽ được tải lên repository GitHub. Có thể truy cập trang repository trên GitHub để xem các file đã được đẩy lên.

Hiển thị file HTML như một trang web thông qua GitHub:

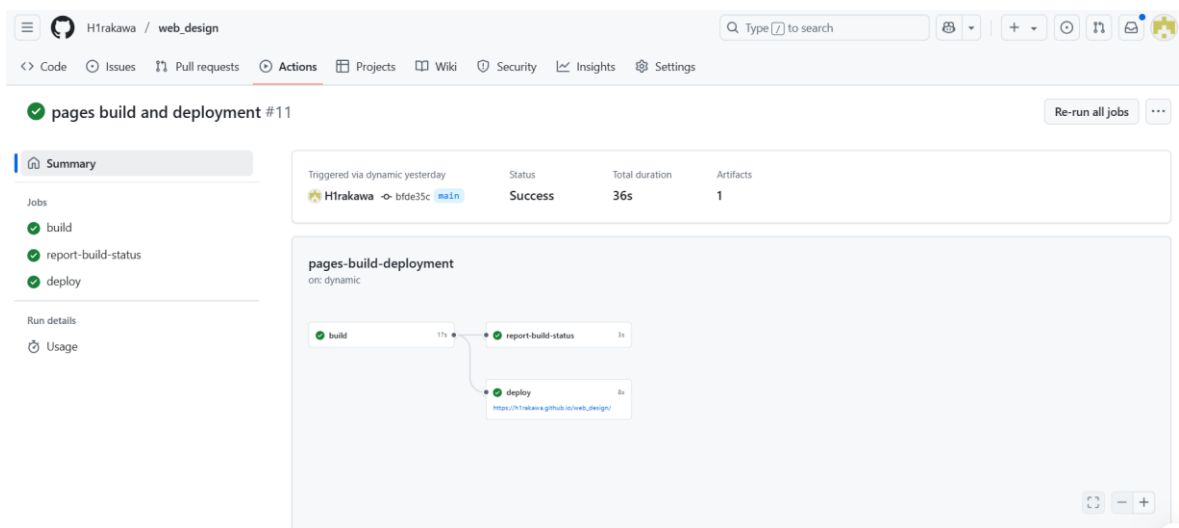
Để file HTML có thể chạy (hiển thị như một trang web) thông qua GitHub, cần sử dụng **GitHub Pages**. GitHub Pages là một dịch vụ miễn phí của GitHub cho phép bạn tạo các trang web tĩnh trực tiếp từ các file HTML, CSS và JavaScript trong repository của bạn.

Đầu tiên phải đảm bảo file HTML đã được đẩy lên GitHub. Hãy chắc chắn rằng file HTML chính (thường là index.html) nằm ở thư mục gốc của repository hoặc trong một thư mục cụ thể được chỉ định sau này.

Kích hoạt và cấu hình GitHub Pages:

- **Truy cập repository của bạn trên GitHub:** Mở trình duyệt web và đi đến trang repository muốn hiển thị trang web HTML.
- **Vào tab “Settings”:** Ở thanh menu ngang phía trên repository (gần các tab “Code”, “Issues”, “Pull requests”), nhấp vào tab “Settings”.
- **Tìm mục “Pages” ở thanh bên trái:** Cuộn xuống dưới trong thanh bên trái cho đến khi bạn thấy mục “Code and automation” và nhấp vào “Pages”.
- **Chọn nguồn (Source) cho GitHub Pages:** Trong phần “Build and deployment”, bạn sẽ thấy một tùy chọn “Source”. Nhấp vào menu thả xuống và chọn một trong các tùy chọn sau:
 - **Deploy from a branch:** Đây là tùy chọn phổ biến nhất cho các trang web tĩnh đơn giản. Khi bạn chọn tùy chọn này, bạn sẽ thấy một menu thả xuống khác để chọn “Branch”. Thông thường, bạn sẽ chọn nhánh **main** (hoặc master nếu repository của bạn vẫn sử dụng nhánh mặc định cũ). Bạn cũng có thể chọn một thư mục cụ thể bên trong nhánh đó (ví dụ: /docs) nếu file index.html của bạn nằm trong thư mục đó.

- **GitHub Actions:** Tùy chọn này phù hợp hơn cho các trang web phức tạp hơn cần quy trình build tùy chỉnh (ví dụ: sử dụng các static site generator như Jekyll, Hugo). Nếu bạn chỉ có file HTML, CSS và JavaScript tĩnh, bạn có thể không cần tùy chọn này.
- **Chọn nhánh (Branch) (nếu bạn chọn “Deploy from a branch”):** Nếu đã chọn “Deploy from a branch”, nhấp vào menu thả xuống **“Branch”** và chọn nhánh chứa file HTML (thường là main). Nếu file index.html của bạn nằm trong thư mục con, có thể chỉ định thư mục đó trong tùy chọn **“Folder”** bên dưới. Chọn / (root) nếu file index.html nằm ở thư mục gốc của repository.
- **Nhấp vào nút “Save”:** Sau khi đã chọn nguồn và nhánh (nếu cần), hãy nhấp vào nút **“Save”**.



Hình 3. 8: Github sau khi hoàn tất build môi trường chạy html

Truy cập trang web. Sau khi lưu cài đặt GitHub Pages, GitHub sẽ tự động build và triển khai trang web. Quá trình này có thể mất vài phút.

Có thể tìm thấy URL của trang web GitHub Pages của mình trong phần **“GitHub Pages”** trên trang **“Settings”** của repository. URL thường có dạng:

`https://<Username_github>.github.io/<Repository_name>/`

Bảng 3. 6: URL trang web chạy file html trên github

- Thay <Your_username_github> bằng tên người dùng GitHub.
- Thay <Your_repository_name> bằng tên repository của bạn.

Nếu file HTML chính của bạn là “index.html” và nó nằm ở thư mục gốc của repository, trang web của bạn sẽ hiển thị khi bạn truy cập URL trên. Nếu file chính của bạn có tên khác (ví dụ: “home.html”), bạn sẽ cần thêm tên file vào URL:

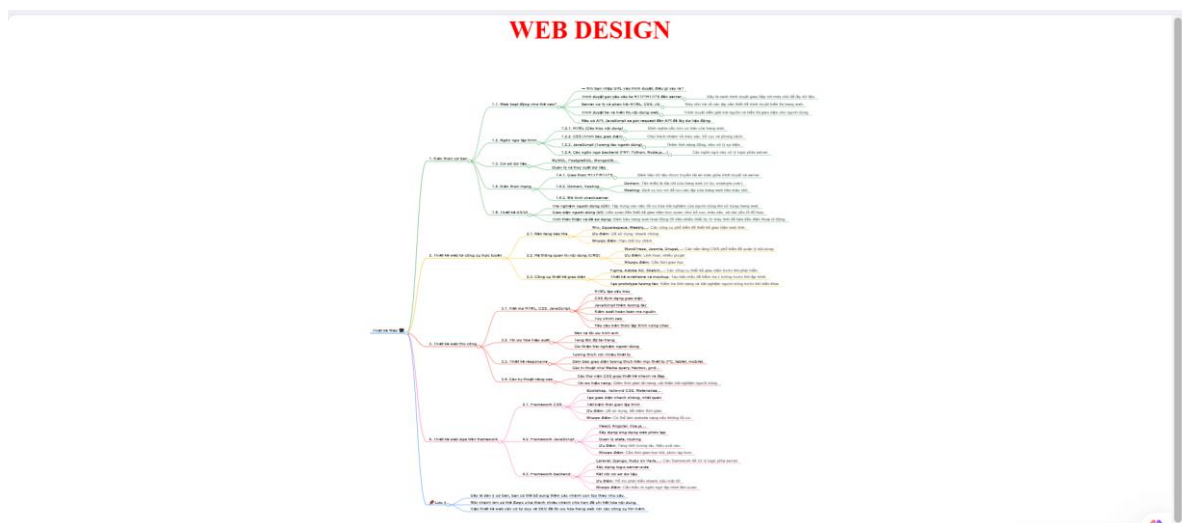
```
https:// Username_github>.github.io/<Repository_name>/ home.html
```

Bảng 3. 7: URL khi không phải file index trên github

Lưu ý:

- **Tên file chính:** GitHub Pages mặc định tìm kiếm file “index.html” (hoặc “index.md”, “README.md”) làm trang chủ của trang web. Nếu file HTML chính của bạn có tên khác, bạn sẽ cần truy cập nó trực tiếp bằng URL có tên file.
- **Public Repository:** Để sử dụng GitHub Pages cho các trang web miễn phí, repository của bạn phải là **public**.
- **Thời gian triển khai:** Có thể mất một chút thời gian (thường là vài phút) để GitHub Pages build và triển khai trang web của bạn sau khi bạn thay đổi cài đặt hoặc đẩy code mới. Hãy kiên nhẫn nếu trang web chưa hiển thị ngay lập tức.

Bằng cách làm theo các bước trên, file HTML có thể chạy và hiển thị như một trang web thông qua GitHub Pages.



Hình 3. 9: Kết quả chạy file html markmap trên môi trường của Github
Đường dẫn để xem markmap: https://hlrakawa.github.io/web_design/

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

Sau quá trình triển khai đề tài “Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web”, các kết quả nghiên cứu đạt được như sau:

Nắm vững kiến thức về Markdown và Markmap

- Đề tài đã giúp làm rõ cú pháp, đặc điểm và cách thức hoạt động của Markdown, từ đó người thực hiện có thể soạn thảo tài liệu một cách nhanh chóng, rõ ràng và có cấu trúc.
- Qua việc tìm hiểu và thực hành, Markmap được chứng minh là công cụ hữu ích để chuyển đổi nội dung Markdown thành sơ đồ tư duy trực quan, giúp trực quan hóa kiến thức và hỗ trợ quá trình học tập hiệu quả.

Triển khai thành công sơ đồ tư duy cho môn Thiết kế web:

- Sản phẩm demo đã được xây dựng dựa trên những nội dung thiết yếu của môn Thiết kế web, bao gồm các kiến thức về HTML, CSS, JavaScript, phương pháp thiết kế web thủ công và dựa trên framework, cũng như việc sử dụng các công cụ trực tuyến.
- Sơ đồ tư duy tạo ra không chỉ thể hiện cấu trúc hệ thống của môn học mà còn mang tính tương tác cao, cho phép người dùng dễ dàng nắm bắt mối liên hệ giữa các phần kiến thức.

Cài đặt môi trường làm việc và tích hợp các công cụ hỗ trợ:

- Môi trường làm việc với Visual Studio Code đã được cấu hình thành công với các tiện ích mở rộng như Markdown Preview Enhanced và Markmap VS Code Extension, tạo điều kiện thuận lợi cho việc soạn thảo và xem trước sơ đồ tư duy.
- Việc tích hợp các công cụ hỗ trợ trực tuyến (như website Markmap) mở rộng khả năng chia sẻ và trình bày sơ đồ tư duy dưới dạng HTML, thuận tiện cho việc báo cáo và giao lưu kiến thức.

Cài đặt môi trường làm việc và tích hợp các công cụ hỗ trợ:

- Qua quá trình thực hiện đề tài, người thực hiện nhận thấy việc sử dụng Markdown kết hợp với Markmap đã giúp hệ thống hóa và trình bày kiến thức một cách trực quan, giảm bớt thời gian tìm kiếm thông tin và hỗ trợ việc học tập, ôn tập một cách hiệu quả.
- Sự phản hồi từ giảng viên và bạn học cho thấy sản phẩm không chỉ có giá trị trong việc học tập mà còn có tiềm năng ứng dụng trong các dự án và bài giảng tương lai.

Những kết quả trên khẳng định rằng việc sử dụng Markdown và Markmap đã góp phần nâng cao hiệu quả truyền đạt và hệ thống hóa kiến thức, đồng thời đem lại trải nghiệm học tập và làm việc hiện đại, sáng tạo cho người dùng. Bạn có thể bổ sung thêm các số liệu, hình ảnh minh họa kết quả (nếu có) để báo cáo trở nên sống động và thuyết phục hơn.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận:

Qua quá trình nghiên cứu và thực hiện đề tài "Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web", người thực hiện đã tích lũy được nhiều kiến thức và kỹ năng bổ ích. Đề tài không chỉ giúp hiểu rõ hơn về cú pháp, đặc trưng và ứng dụng của ngôn ngữ Markdown, mà còn khám phá được cách sử dụng công cụ Markmap để trực quan hóa nội dung học tập một cách sinh động và khoa học.

Việc áp dụng Markmap để xây dựng sơ đồ tư duy cho môn Thiết kế web đã chứng minh rằng, việc trực quan hóa kiến thức mang lại nhiều lợi ích: dễ hệ thống thông tin, dễ ghi nhớ, và tạo sự liên kết logic giữa các phần nội dung. Bên cạnh đó, việc thiết lập môi trường làm việc trên Visual Studio Code cùng các tiện ích hỗ trợ cũng giúp quy trình soạn thảo và trình bày trở nên thuận tiện, chuyên nghiệp hơn.

Mặc dù đề tài đã đạt được những kết quả như mong đợi, tuy nhiên trong quá trình thực hiện vẫn còn những hạn chế nhất định, như phạm vi nội dung còn gói gọn, chưa khai thác sâu hơn vào các tính năng mở rộng của Markmap. Trong tương lai, đề tài có thể được phát triển thêm bằng cách tích hợp nhiều công cụ hỗ trợ khác và mở rộng ứng dụng vào nhiều môn học khác nhau.

Tóm lại, việc tìm hiểu và vận dụng Markdown và Markmap không chỉ giúp nâng cao hiệu quả học tập mà còn góp phần trang bị cho sinh viên kỹ năng làm việc hiện đại, thích ứng tốt với yêu cầu công việc thực tế trong lĩnh vực công nghệ thông tin.

Hướng phát triển:

Dựa trên những kết quả đạt được, đề tài "Tìm hiểu ngôn ngữ Markdown và công cụ Markmap để xây dựng sơ đồ tư duy cho môn học Thiết kế web" có thể được mở rộng và phát triển theo các hướng sau:

- **Mở rộng nội dung sơ đồ tư duy:** Bổ sung thêm nhiều chủ đề chi tiết hơn liên quan đến thiết kế web như: bảo mật web cơ bản, tối ưu trải

nghiệm người dùng (UX), thiết kế giao diện người dùng (UI), các xu hướng thiết kế web hiện đại (Progressive Web App, Web 3.0...).

- **Ứng dụng Markmap trong nhiều lĩnh vực khác:** Không chỉ giới hạn ở môn Thiết kế web, công cụ Markmap còn có thể áp dụng để xây dựng sơ đồ tư duy cho các môn học khác như Lập trình, Cơ sở dữ liệu, Quản lý dự án, hoặc dùng trong việc quản lý kiến thức cá nhân.
- **Tích hợp nhiều công cụ hỗ trợ khác:** Kết hợp Markmap với các công cụ khác như Obsidian, Notion, hoặc các nền tảng lưu trữ mã nguồn như GitHub Pages để chia sẻ và quản lý sơ đồ tư duy online một cách chuyên nghiệp hơn.
- **Nâng cao kỹ năng trình bày và tối ưu hóa sơ đồ:** Tìm hiểu thêm các kỹ thuật tối ưu sơ đồ như rút gọn nội dung, thiết kế bố cục hợp lý, sử dụng màu sắc phân loại để giúp sơ đồ tư duy trực quan, dễ hiểu và thẩm mỹ hơn.
- **Phát triển ứng dụng thực tế:** Xây dựng một website cá nhân sử dụng Markdown và Markmap để lưu trữ, trình bày các dự án học tập hoặc các bài nghiên cứu, từ đó tăng cường kỹ năng thực hành và xây dựng hồ sơ năng lực cá nhân.

DANH MỤC TÀI LIỆU THAM KHẢO

1. Matt Cone, Markdown Guide, June 27, 2020, Independently Published, pp. 4 – 19.
2. Nehal Khan, Markdown: A Complete Guide, December 6, 2023, Independently Published, website
<https://bookdown.org/yihui/rmarkdown/>
3. Yihui Xie, J.J. Allaire, Garrett Grolmund, R Markdown: The Definitive Guide, July 17, 2018, Chapman & Hall/CRC, Boca Raton, Florida, Hoa Kỳ, pp. 11 – 30.
4. Thomas Mailund, Introducing Markdown and Pandoc: Using Markup Language and Document Converter, August 20, 2019, Apress, Berkeley, California, Hoa Kỳ, pp. 1 - 40

PHỤ LỤC

Danh sách từ viết tắt:

- **VSCode:** Visual Studio Code - Trình soạn thảo mã nguồn được phát triển bởi Microsoft.
- **URL: Uniform Resource Locator** - Định vị tài nguyên thống nhất.
- **TOC: Table of Contents** - Mục lục.
- **MD: Markdown** - ngôn ngữ đánh dấu nhanh gọn.
- **HTML: HyperText Markup Language** - Ngôn ngữ đánh dấu tiêu chuẩn tạo cấu trúc cho một trang web

Mã nguồn file Markdown:

```
---
title: Thiết kế Web 🖥️
markmap:
  colorFreezeLevel: 2
  color:
    - "#4287f5"
    - "#34a853"
    - "#fbbc05"
    - "#ea4335"
    - "#ff69B4"
---

# **Thiết kế Web** 🖥️
## **1. Kiến thức cơ bản**
### **1.1. Web hoạt động như thế nào?**
#### **→ Khi bạn nhập URL vào trình duyệt, điều gì xảy ra?**
#### **Trình duyệt gửi yêu cầu từ HTTP/HTTPS đến server.**
- Đây là cách trình duyệt giao tiếp với máy chủ để lấy dữ liệu.
#### **Server xử lý và phản hồi HTML, CSS, JS.**
- Máy chủ trả về các tệp cần thiết để trình duyệt hiển thị trang web.
```

Trình duyệt tải và hiển thị nội dung web.

- Trình duyệt diễn giải mã nguồn và hiển thị giao diện cho người dùng.

Nếu có API, JavaScript sẽ gửi request đến API để lấy dữ liệu động.

1.2. Ngôn ngữ lập trình

1.2.1. HTML (Cấu trúc nội dung)

- Định nghĩa cấu trúc cơ bản của trang web.

1.2.2. CSS (Trình bày giao diện)

- Chịu trách nhiệm về màu sắc, bố cục và phong cách.

1.2.3. JavaScript (Tương tác người dùng)

- Thêm tính năng động, như xử lý sự kiện.

1.2.4. Các ngôn ngữ backend (PHP, Python, Node.js,...)

- Các ngôn ngữ này xử lý logic phía server.

1.3. Cơ sở dữ liệu

MySQL, PostgreSQL, MongoDB,...

Quản lý và truy xuất dữ liệu

1.4. Kiến thức mạng

1.4.1. Giao thức HTTP/HTTPS

- Đảm bảo dữ liệu được truyền tải an toàn giữa trình duyệt và server

1.4.2. Domain, hosting

- **Domain:** Tên miền là địa chỉ của trang web (ví dụ: example.com).

- **Hosting:** Dịch vụ lưu trữ để lưu các tệp của trang web trên máy chủ.

1.4.3. Mô hình client-server

1.5. Thiết kế UX/UI

Trải nghiệm người dùng (UX): Tập trung vào việc tối ưu hóa trải nghiệm của người dùng khi sử dụng trang web.

Giao diện người dùng (UI): Liên quan đến thiết kế giao diện trực quan, như bố cục, màu sắc, và các yếu tố đồ họa.

Tính thân thiện và dễ sử dụng: Đảm bảo trang web hoạt động tốt trên nhiều thiết bị, từ máy tính để bàn đến điện thoại di động.

2. Thiết kế web từ công cụ trực tuyến

2.1. Nền tảng kéo thả

Wix, Squarespace, Weebly,...: Các công cụ phổ biến để thiết kế giao diện web tĩnh.

Ưu điểm: Dễ sử dụng, nhanh chóng

Nhược điểm: Hạn chế tùy chỉnh

2.2. Hệ thống quản trị nội dung (CMS)

WordPress, Joomla, Drupal,...: Các nền tảng CMS phổ biến để quản lý nội dung.

Ưu điểm: Linh hoạt, nhiều plugin

Nhược điểm: Cần thời gian học

2.3. Công cụ thiết kế giao diện

Figma, Adobe XD, Sketch,...: Các công cụ thiết kế giao diện trước khi phát triển.

Thiết kế wireframe và mockup: Tạo bản mẫu để kiểm tra ý tưởng trước khi lập trình.

Tạo prototype tương tác: Kiểm tra tính năng và trải nghiệm người dùng trước khi triển khai.

3. Thiết kế web thủ công

3.1. Viết mã HTML, CSS, JavaScript

HTML tạo cấu trúc

CSS định dạng giao diện

JavaScript thêm tương tác

Kiểm soát hoàn toàn mã nguồn

Tùy chỉnh cao

Yêu cầu kiến thức lập trình vững chắc

3.2. Tối ưu hóa hiệu suất

Nén và tối ưu hình ảnh

Tăng tốc độ tải trang

Cải thiện trải nghiệm người dùng

3.3. Thiết kế responsive

Tương thích với nhiều thiết bị

****Đảm bảo giao diện tương thích trên mọi thiết bị (PC, tablet, mobile).****

****Các kỹ thuật như Media query, flexbox, grid...****

****3.4. Các kỹ thuật nâng cao****

****Các thư viện CSS giúp thiết kế nhanh và đẹp.****

****Tối ưu hiệu năng:**** Giảm thời gian tải trang, cải thiện trải nghiệm người dùng.

****4. Thiết kế web dựa trên framework****

****4.1. Framework CSS****

****Bootstrap, Tailwind CSS, Materialize,...**

****Tạo giao diện nhanh chóng, nhất quán****

****Tiết kiệm thời gian lập trình****

****Ưu điểm:**** Dễ sử dụng, tiết kiệm thời gian.

****Nhược điểm:**** Có thể làm website nặng nếu không tối ưu.

****4.2. Framework JavaScript****

****React, Angular, Vue.js,...**

****Xây dựng ứng dụng web phức tạp****

****Quản lý state, routing****

****Ưu điểm:**** Tăng tính tương tác, hiệu quả cao.

****Nhược điểm:**** Cần thời gian học hỏi, phức tạp hơn.

****4.3. Framework backend****

****Laravel, Django, Ruby on Rails,...** Các framework để xử lý logic phía server.

****Xây dựng logic server-side****

****Kết nối cơ sở dữ liệu****

****Ưu điểm:**** Hỗ trợ phát triển nhanh, bảo mật tốt.

****Nhược điểm:**** Cần hiểu rõ ngôn ngữ lập trình liên quan.

****📌 Lưu ý****

****Đây là dàn ý cơ bản, bạn có thể bổ sung thêm các nhánh con tùy theo nhu cầu.****

Mỗi nhánh lớn có thể được chia thành nhiều nhánh nhỏ hơn để chi tiết hóa nội dung.

Việc thiết kế web cần có tư duy về SEO để tối ưu hóa trang web với các công cụ tìm kiếm.