

Claude từ người mới đến chuyên gia

Chương trình đầy đủ, không bỏ sót một mảng nào: từ lần đầu tải ứng dụng, qua kết nối dữ liệu thật, đến điều khiển Claude Code và tự tay xây skill — agent — công cụ riêng cho cả một đội AI.

4 tầng năng lực · 22 bài học

Lộ trình: 8 tuần

Nền tảng: macOS 26

Đích đến: tự vận hành đội AI

BẢN ĐỒ NĂNG LỰC

Bốn tầng — mỗi tầng là một cấp thành thạo trọn vẹn

Học tuần tự từ trái sang phải. Xong tầng nào dùng được ngay tầng đó; đi hết bốn tầng là làm chủ toàn bộ hệ sinh thái Claude.

Tầng 01

Người dùng

Trò chuyện, prompt, đa phương tiện, Artifacts. Giải quyết công việc thật mà không cần kỹ thuật.

CLAUDE DESKTOP · MOBILE

Tầng 02

Người kết nối

Nói dữ liệu thật, tìm kiếm web, suy nghĩ sâu, chạy code phân tích, điều khiển trình duyệt.

CONNECTORS · RESEARCH

Tầng 03

Vận hành viên

Terminal, Claude Code, CLAUDE.md, skills, agents, MCP, Git, deploy, tự động hoá theo lịch.

CLAUDE CODE (CLI)

Tầng 04

Chuyên gia

Tự xây skill/agent/MCP riêng, điều phối nhiều agent, bảo mật, tối ưu chi phí & mô hình.

KIẾN TRÚC SƯ AI

Kết thúc tầng: học viên tự cài app, viết prompt sắc, làm việc với ảnh–file–giọng nói, và tạo ra sản phẩm hoàn chỉnh (tài liệu, trang web mini, bảng tính) ngay trong app.

1.1 Tài khoản, gói cước & cài đặt

✓ App chạy trên cả máy tính lẫn điện thoại, hiểu mình đang ở gói nào.

- Tải **Claude Desktop** tại `claude.ai/download` · app điện thoại: “Claude by Anthropic” trên App Store / CH Play.
- Đăng nhập một tài khoản — đồng bộ mọi thiết bị và cả bản web `claude.ai`.
- Gói: **Free** (thử) → **Pro** (làm việc hằng ngày) → **Max** (dùng nặng + mở khoá Claude Code ở tầng 3).
- Phân biệt **Claude.ai / app** (trò chuyện) với **Claude Code** (làm sản phẩm trong terminal) — sẽ học ở tầng 3.

Thực hành: cài xong, gõ câu đầu tiên và gửi kèm 1 tấm ảnh chụp màn hình để thử.

1.2 Giao diện, Projects & chọn model

✓ Biết chỗ gõ, chỗ lưu, khi nào dùng model nào.

- Ô chat nhận chữ, ảnh, PDF/Word/Excel bằng kéo–thả. Lịch sử nằm cột trái.
- **Projects:** gom một chủ đề — nạp tài liệu nền + “chỉ dẫn tùy chỉnh” để Claude nhớ ngữ cảnh xuyên suốt.
- Chọn model: **Opus** (thông minh nhất, việc khó) · **Sonnet** (nhanh–mạnh cân bằng) · **Haiku** (nhẹ, tức thì).
- Quản lý hội thoại: chỉnh sửa câu đã gửi, rẽ nhánh thử hướng khác, đổi tên & ghim.

Thực hành: tạo Project “Công việc của tôi”, nạp 1 file nền + viết chỉ dẫn tùy chỉnh về giọng văn.

1.3 Ra lệnh cơ bản — công thức 4 phần

✓ Không còn “hỏi cho có”, prompt nào cũng có cấu trúc.

- **Vai trò + Bối cảnh + Nhiệm vụ + Định dạng đầu ra.**
- VD: “Đóng vai chuyên gia dinh dưỡng nhi. Con tôi 3 tuổi biếng ăn. Soạn thực đơn 3 ngày. Trình bày dạng bảng.”
- Nói rõ độ dài, giọng văn, đối tượng đọc, điều cần tránh.
- Chưa ưng thì **sửa tại chỗ**: “ngắn hơn”, “thân mật hơn”, “thêm ví dụ” — không viết lại từ đầu.

Thực hành: mỗi người viết 1 prompt cho việc thật của mình theo đúng 4 phần.

1.4 Prompt nâng cao

✓ Ép Claude cho ra chất lượng cao ổn định.

- **Cho ví dụ mẫu** (few-shot): dán 1–2 bài mẫu “kiểu tôi muốn” để Claude bắt chước văn phong.
- **Yêu cầu suy nghĩ từng bước** cho bài toán khó: “Phân tích trước, kết luận sau.”
- **Chia việc lớn thành chuỗi**: dàn ý → viết → biên tập, mỗi bước một lượt.
- **Ràng buộc rõ**: số từ, cấm sáo rỗng, phải có số liệu, đúng đối tượng.

Thực hành: đưa 1 bài viết mẫu, yêu cầu Claude viết bài mới “cùng giọng, khác chủ đề”.

1.5 Làm việc đa phương tiện

✓ Không chỉ chữ — dùng được ảnh, tài liệu, giọng nói.

- **Ảnh**: chụp màn hình lỗi, hoá đơn, biểu đồ → nhờ Claude đọc, giải thích, bóc số liệu.
- **Nhiều file**: thả cả xấp → so sánh, tổng hợp, tìm mâu thuẫn giữa chúng.
- **Giọng nói** (mobile): nói thay vì gõ; đọc file ghi âm → tóm tắt.
- **Web**: dán link, Claude đọc trang rồi tóm tắt / trích dẫn.

Thực hành: chụp một bảng số liệu giấy, nhờ Claude gõ lại thành bảng và tính tổng.

1.6 Artifacts — tạo ra sản phẩm

✓ Ra khỏi ô chat: có tài liệu / trang web / bảng tải về được.

- Yêu cầu tạo **tài liệu, bảng tính, trang web mini, sơ đồ** → hiện ở khung Artifact bên phải.
- Sửa trực tiếp bằng lời: "đổi màu tiêu đề", "thêm cột giá", "bỏ đoạn cuối".
- Tải về hoặc chia sẻ link; dùng lại như mẫu cho lần sau.

Thực hành: "Tạo cho tôi 1 trang giới thiệu dịch vụ 1 màn hình" rồi tải về.

TẦNG 02 · NGƯỜI DÙNG → NGƯỜI KẾT NỐI

Nối Claude với dữ liệu & công cụ thật

Kết thúc tầng: Claude làm việc trực tiếp trên email, tài liệu, thiết kế của bạn; tự tìm thông tin trên web; suy nghĩ sâu cho bài khó; và chạy được tính toán/biểu đồ ngay trong hội thoại.

2.1 Connectors — cắm là dùng

✓ Claude đọc & thao tác trên Gmail / Drive / Canva... khi được cho phép.

- **Settings → Connectors**, bật kết nối cần dùng: Gmail, Google Drive, Calendar, Canva, Notion, Slack...
- Mỗi kết nối phải **tự bấm đồng ý** trong cửa sổ đăng nhập của chính dịch vụ đó.
- Việc làm được: "Soạn nháp trả lời khách trong hộp thư", "Tìm hợp đồng tháng 6 trên Drive".

An toàn: gửi email, xoá dữ liệu, đổi quyền chia sẻ — Claude luôn **hỏi xác nhận** trước. Chỉ cấp quyền cho kết nối bạn thật sự dùng.

2.2 Tìm kiếm web & nghiên cứu

✓ Câu trả lời có dẫn nguồn, cập nhật, không “bịa”.

- Bật **Web Search** để Claude tra thông tin mới và trích link nguồn.
- Chế độ **Research**: giao một câu hỏi lớn, Claude tự tra nhiều nguồn rồi tổng hợp thành báo cáo.
- Luôn yêu cầu “kèm nguồn” cho các con số quan trọng.

Thực hành: “Nghiên cứu giúp tôi xu hướng thị trường X năm nay, kèm nguồn.”

2.3 Suy nghĩ sâu & công cụ phân tích

✓ Bài khó ra kết quả chắc; số liệu ra biểu đồ ngay trong chat.

- **Extended thinking**: bật cho bài toán logic/lập kế hoạch phức tạp — Claude “nghĩ kỹ” trước khi trả lời.
- **Analysis tool**: Claude chạy tính toán thật trên file Excel/CSV bạn đưa, vẽ biểu đồ, kiểm tra số liệu.

Thực hành: thả file doanh thu CSV, yêu cầu “vẽ biểu đồ theo tháng và chỉ ra tháng bất thường”.

2.4 Claude trong trình duyệt

✓ Claude nhìn & thao tác trên đúng trang web bạn đang mở.

- Tiện ích **Claude for Chrome**: đọc trang hiện tại, điền form, tóm tắt, hỗ trợ thao tác.
- Dùng cho: rà soát trang web, so sánh sản phẩm, hỗ trợ nhập liệu lặp đi lặp lại.

Cảnh giác: đừng để Claude tự bấm link lạ hay nhập thông tin nhạy cảm từ email/tin nhắn không rõ nguồn — luôn kiểm tra trước.

Kết thúc tầng: học viên dựng xong nền phần mềm, dùng thành thạo Claude Code để đọc/sửa dự án, cấu hình CLAUDE.md, gọi skill-agent-MCP, làm việc với Git, deploy web và đặt tự động hoá chạy nền.

3.1 Terminal vỡ lòng (cho người chưa từng dùng)

✓ Hết sợ “màn hình đen”, gõ được các lệnh đi lại cơ bản.

- Mở **Terminal** (Spotlight → gõ “Terminal”). Đây là nơi ra lệnh bằng chữ.
- Lệnh sống còn: `pwd` (đang ở đâu) · `ls` (liệt kê) · `cd` (đi vào thư mục) · `mkdir` (tạo thư mục).
- Kéo-thả một thư mục vào Terminal để lấy đường dẫn nhanh.
- Không thuộc lệnh? Hỏi thẳng Claude: “Lệnh gì để làm X trên Mac?”.

Thực hành: tạo thư mục `~/hoc-claude` rồi `cd` vào đó.

3.2 Dựng nền phần mềm (một lần duy nhất)

✓ Máy đủ bộ đồ nghề để chạy Claude Code.

- **Homebrew** — cửa hàng cài phần mềm cho Mac bằng lệnh (nền của mọi thứ).
- **Node.js** — bộ máy chạy Claude Code · **Git** — quản lý phiên bản an toàn.

```
# 1 · Cài Homebrew
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

# 2 · Cài Node.js + Git
brew install node git

# 3 · Cài Claude Code
npm install -g @anthropic-ai/claude-code
```

3.3 Khởi động & ra lệnh cho Claude Code

✓ Gõ `claude` là có trợ lý làm việc ngay trên dự án.

- Vào thư mục dự án: `cd đường-dẫn` → gõ `claude` → đăng nhập bằng gói Pro/Max.
- Ra lệnh bằng tiếng Việt: “Đọc thư mục này và giải thích dự án làm gì.”
- Claude Code đọc file, sửa code, chạy lệnh, deploy — **luôn xin phép** trước khi đổi thứ quan trọng.
- **Plan mode**: bảo Claude “lập kế hoạch trước, chưa sửa” để duyệt hướng đi.

3.4 CLAUDE.md, lệnh gạch chéo & chế độ quyền

✓ Điều khiển Claude Code chuyên nghiệp, không lặp lại chính mình.

- **CLAUDE.md** — “luật chơi” của dự án: Claude tự đọc mỗi phiên (giọng văn, quy tắc, ngữ cảnh).
- Lệnh gạch chéo hay dùng: `/init`, `/clear`, `/compact`, `/model`, `/agents`, `/mcp`.
- Chế độ quyền: hỏi từng bước ↔ tự động chấp nhận ↔ chỉ lập kế hoạch.
- **Bộ nhớ**: ghi lại sở thích/quyết định để phiên sau không phải nói lại.

Thực hành: chạy `/init` để Claude tự sinh file CLAUDE.md cho dự án.

3.5 Skills • Agents • MCP

✓ Một câu lệnh gọi ra cả đội chuyên gia và công cụ thật.

- **Skill** — gói kỹ năng gọi bằng `/tên-skill` (VD dựng landing page, viết bài chuẩn SEO).
- **Agent** — chuyên gia ảo lãnh một mảng (SEO, thiết kế, phân tích) tự làm trọn việc.
- **MCP** — cắm công cụ ngoài (Facebook, Vercel, Drive, Canva) để Claude thao tác thật.

Thực hành: gọi 1 skill có sẵn cho ra 1 sản phẩm thật rồi lấy link.

3.6 Git, Deploy & tự động hoá

✓ Đưa sản phẩm lên mạng và cho việc tự chạy theo lịch.

- **Git**: Claude Code tạo commit gọn gàng, giữ lịch sử để lùi lại khi cần.
- **Deploy**: đẩy web lên Vercel → có link live chia sẻ được.
- **Tự động hoá**: đặt việc chạy nền theo giờ (đăng bài, báo cáo quảng cáo, sao lưu ảnh).

Thực hành: deploy một trang lên Vercel và dán link vào nhóm.

TẦNG 04 · VẬN HÀNH VIÊN → CHUYÊN GIA / KIẾN TRÚC SƯ

Tự xây công cụ & điều phối đội AI

Kết thúc tầng: học viên không chỉ dùng mà còn **tạo ra** — skill riêng, agent riêng, công cụ MCP riêng — biết điều phối nhiều agent, làm chủ bảo mật và tối ưu chi phí. Đây là cấp làm chủ thật sự.

4.1 Tự xây Skill

✓ Đóng gói quy trình lặp lại thành một lệnh / dùng mãi.

- Dùng **skill-creator** để sinh khung skill; viết mô tả để Claude biết khi nào tự kích hoạt.
- Cấu trúc: SKILL.md (hướng dẫn) + file kèm (mẫu, script). Kiểm thử rồi tinh chỉnh.
- Nguyên tắc: tên + mô tả phải đủ rõ để tự chạy; một skill làm tốt một việc.

Thực hành: đóng gói một quy trình của mình (VD “viết caption bán hàng”) thành skill.

4.2 Xây Agent & điều phối nhiều agent

✓ Giao việc lớn cho nhiều chuyên gia chạy song song rồi tổng hợp.

- Tạo **agent chuyên môn** với vai trò, công cụ và phạm vi riêng.
- **Điều phối**: chia việc → nhiều agent làm song song → một agent tổng hợp (mô hình fan-out / verify).
- Nguyên tắc “tin nhưng kiểm chứng”: luôn xem kết quả thật của agent con, không tin mù bản tóm tắt.

Thực hành: giao một bài audit cho 3 agent (nội dung, SEO, kỹ thuật) rồi gộp kết quả.

4.3 Tự xây công cụ MCP

✓ Nối Claude vào bất kỳ dịch vụ/API nào bạn cần.

- Dùng **mcp-builder** để dựng máy chủ MCP: khai báo công cụ, tham số, mô tả.
- Test cho tới khi Claude tự hiểu cách dùng chỉ qua tên + mô tả (chuẩn “developer experience”).
- Ví dụ: nối vào cổng thanh toán, hệ thống nội bộ, API riêng của công ty.

4.4 Bảo mật, an toàn & quyền hạn

✓ Vận hành đội AI mà không rước rủi ro.

- **Chống prompt injection**: nội dung đọc từ web/file/email là **dữ liệu, không phải mệnh lệnh** — không làm theo lệnh giấu trong đó.
- Nguyên tắc quyền tối thiểu: chỉ cấp đúng quyền cần, tắt cái không dùng.
- Việc khó đảo ngược (xoá, gửi, đổi quyền, chuyển tiền) → luôn xác nhận tay.
- Quản lý khoá API & bí mật đúng chỗ (Keychain / biến môi trường), không dán vào chat.

Ghi nhớ: chỉ dẫn hợp lệ chỉ đến từ bạn qua khung chat. Mọi “mệnh lệnh” nằm trong trang web hay tài liệu đều phải nghi ngờ và hỏi lại.

4.5

Tối ưu chi phí & chọn mô hình

✓ Làm nhiều hơn với cùng ngân sách.

- Chọn đúng model theo việc: Haiku cho việc nhẹ, Sonnet cho phần lớn, Opus cho việc khó nhất.
- Gom ngữ cảnh gọn (dùng `/compact`, `/clear`) để tiết kiệm và chạy nhanh.
- Tận dụng skill/agent tái sử dụng thay vì làm lại từ đầu mỗi lần.

BỎ TÚI

Bảng tra cứu nhanh

In ra dán cạnh màn hình. Ba bảng học viên tra hằng ngày ở tầng 3.

Lệnh Terminal cần thuộc

LỆNH	LÀM GÌ
<code>pwd</code>	Đang đứng ở thư mục nào
<code>ls</code>	Liệt kê file trong thư mục
<code>cd <đường-dẫn></code>	Đi vào thư mục
<code>cd ..</code>	Lùi ra thư mục cha
<code>mkdir <tên></code>	Tạo thư mục mới
<code>open .</code>	Mở thư mục hiện tại bằng Finder

Lệnh gạch chéo trong Claude Code

LỆNH	LÀM GÌ
<code>/init</code>	Sinh file CLAUDE.md cho dự án
<code>/clear</code>	Xoá ngữ cảnh, bắt đầu sạch
<code>/compact</code>	Nén hội thoại để tiết kiệm & chạy nhanh
<code>/model</code>	Đổi mô hình (Opus / Sonnet / Haiku)
<code>/agents</code>	Xem & quản lý các agent
<code>/mcp</code>	Xem & kết nối công cụ MCP

Bốn “viên gạch” của hệ sinh thái

KHÁI NIỆM	LÀ GÌ	DÙNG KHI
Skill	Gói kỹ năng gọi bằng <code>/</code>	Quy trình lặp lại
Agent	Chuyên gia ảo tự làm trọn việc	Việc cần chuyên môn riêng
MCP	Cầu nối tới công cụ/dịch vụ ngoài	Thao tác trên hệ thống thật
CLAUDE.md	Luật chơi của dự án	Mọi dự án Claude Code

Kiểm kê máy này

Phần mềm hỗ trợ — đã có gì, nên thêm gì

Kiểm kê thực tế trên máy Mac hiện tại. Nền tảng đã sẵn sàng; danh sách “nên thêm” giúp làm việc mượt hơn, không bắt buộc.

Nhóm A — Nền tảng bắt buộc

ĐÃ CÓ ĐỦ

PHẦN MỀM	VAI TRÒ	TRẠNG THÁI
Claude Desktop	Ứng dụng chính để dùng Claude	✓ Đã cài
Homebrew 6.0	Trình cài đặt phần mềm cho Mac	✓ Đã cài
Node.js v26	Bộ máy chạy Claude Code	✓ Đã cài
Git 2.50	Quản lý phiên bản dự án	✓ Đã cài
Python 3.14	Chạy script tự động hoá	✓ Đã cài

Nhóm B — Hỗ trợ media & tự động hoá ĐÃ CÓ ĐỦ

PHẦN MỀM	VAI TRÒ	TRẠNG THÁI
ffmpeg	Cắt / ghép / xuất video, âm thanh	✓ Đã cài
yt-dlp	Tải video & phụ đề từ web	✓ Đã cài
whisper-cli	Bóc lời nói ra chữ (tiếng Việt)	✓ Đã cài
jq	Xử lý dữ liệu JSON trong lệnh	✓ Đã cài

Nhóm C — Nên bổ sung để làm mượt hơn TUỶ CHỌN

PHẦN MỀM	VÌ SAO NÊN THÊM	CÀI BẰNG
VS Code	Xem/sửa file Claude Code tạo ra — trực quan hơn terminal	<code>brew install --cask visual-studio-code</code>
GitHub CLI	Đẩy dự án lên GitHub, tạo Pull Request bằng lệnh	<code>brew install gh</code>
iTerm2	Terminal đẹp & mạnh hơn bản mặc định	<code>brew install --cask iterm2</code>
uv	Chạy & quản lý Python nhanh hơn nhiều lần	<code>brew install uv</code>

Lưu ý PATH: nếu gõ `claude` báo “not found”, chạy `which claude` kiểm tra; thường chỉ cần mở lại terminal hoặc thêm đường dẫn npm global vào PATH.

LỘ TRÌNH

Tám tuần — tuần nào cũng ra một sản phẩm

Học đến đâu làm ra sản phẩm đến đó. Hai tuần một tầng, không lý thuyết suông.

Tuần 01 · Tầng 1

Làm chủ ứng dụng

Cài app, prompt 4 phần, Project đầu tiên. → Giải quyết 1 việc thật.

Tuần 02 · Tầng 1

Nâng cao & đa phương tiện

Prompt mẫu, ảnh-file-giọng nói, Artifacts. → 1 trang giới thiệu tự tạo.

Tuần 03 · Tầng 2

Kết nối dữ liệu

Connectors, làm việc trên email/tài liệu thật. → 1 bản tổng hợp từ dữ liệu của mình.

Tuần 04 · Tầng 2

Nghiên cứu & phân tích

Web search, Research, Analysis tool, Chrome. → 1 báo cáo có nguồn + biểu đồ.

Tuần 05 · Tầng 3

Terminal & Claude Code

Dựng nền, lệnh cơ bản, gõ lệnh đầu tiên. → Claude Code giải thích 1 dự án.

Tuần 06 · Tầng 3

Skill · Agent · Deploy

CLAUDE.md, gọi skill/agent, Git, deploy. → 1 web live có link.

Tuần 07 · Tầng 4

Tự xây công cụ

Tạo skill & agent riêng. → 1 skill của chính mình chạy được.

Tuần 08 · Tầng 4

Điều phối & bảo mật

Nhiều agent, MCP riêng, an toàn, tối ưu chi phí. → 1 quy trình tự động hoàn chỉnh.

TỪ ĐIỂN

Thuật ngữ cho người mới

Mọi từ “lạ tai” xuất hiện trong khoá, giải thích một câu.

Prompt

Câu lệnh / yêu cầu bạn gõ cho Claude.

Model

“Bộ não” xử lý — Opus, Sonnet, Haiku khác nhau về sức mạnh & tốc độ.

Project

Không gian gom một chủ đề, giữ ngữ cảnh xuyên suốt.

Artifact

Sản phẩm Claude tạo ra (tài liệu/web/bảng) hiện bên cạnh chat.

Connector

Cầu nối app với dịch vụ như Gmail, Drive, Canva.

Terminal

Cửa sổ ra lệnh cho máy bằng chữ ("màn hình đen").

Claude Code

Phiên bản Claude chạy trong terminal để làm ra sản phẩm.

CLAUDE.md

File "luật chơi" Claude Code tự đọc mỗi dự án.

Skill

Gói kỹ năng gọi nhanh bằng `/tên-skill`.

Agent

Chuyên gia ảo tự làm trọn một mảng việc.

MCP

Chuẩn cắm công cụ ngoài để Claude thao tác thật.

Deploy

Đưa web lên mạng để có link chia sẻ được.

TỐT NGHIỆP

Checklist thành thạo — bốn cấp

Tick đủ 12 ô là làm chủ toàn bộ hệ sinh thái Claude, từ người mới đến chuyên gia.

Cấp 1 · Người dùng

1 Tự cài & đăng nhập Claude trên máy tính và điện thoại.

2 Viết prompt 4 phần và làm việc được với ảnh + file.

3 Tạo Project và xuất một Artifact hoàn chỉnh.

Cấp 2 · Người kết nối

4 Bật Connector và giao cho Claude việc trên dữ liệu của mình.

5 Chạy Research có dẫn nguồn.

6 Dùng Analysis tool vẽ biểu đồ từ file số liệu.

Cấp 3 · Vận hành viên

7 Cài xong nền phần mềm, chạy được lệnh `claude`.

8 Dùng CLAUDE.md và lệnh gạch chéo thành thạo.

9 Gọi skill/agent + deploy 1 sản phẩm có link.

Cấp 4 · Chuyên gia

10 Tự xây 1 skill của riêng mình.

11 Điều phối nhiều agent cho một việc lớn.

12 Áp dụng quy tắc an toàn & tối ưu chi phí khi vận hành.

Ba lỗi thường gặp cần né: ① cấp quyền Connector bừa bãi cho dịch vụ không dùng · ② prompt cụt lùn rồi trách Claude trả lời chung chung · ③ bỏ qua bước dựng nền (Homebrew/Node) rồi thắc mắc sao `claude` không chạy.