

ỨNG DỤNG GENERATIVE AI ĐỂ XÂY DỰNG KHO BÀI TẬP TÌNH HUỐNG ĐA LĨNH VỰC CHO HỌC PHẦN TÍCH THIẾT KẾ HỆ THỐNG THÔNG TIN

Nguyễn Thị Kim Nhung
Khoa Công nghệ thông tin, Đại học Nghệ An

Tóm tắt: Nghiên cứu đề xuất quy trình ứng dụng Prompt Engineering để tự động hóa việc thiết kế Case Study trong giảng dạy Phân tích và thiết kế hệ thống thông tin hướng đối tượng (OOAD). Kết quả thực nghiệm cho thấy giải pháp giúp tiết kiệm 90% thời gian soạn thảo, tạo ra học liệu đa dạng, thực tế và ngăn chặn hiệu quả hành vi sao chép của sinh viên.

Từ khóa: Kỹ thuật thiết kế câu lệnh, Bài tập tình huống OOAD, AI tạo sinh trong giáo dục.

APPLYING GENERATIVE AI TO DEVELOP A MULTIDISCIPLINARY CASE-BASED EXERCISE REPOSITORY FOR THE INFORMATION SYSTEMS ANALYSIS AND DESIGN COURSE

Abstract: This study proposes a workflow for applying Prompt Engineering to automate the design of Case Studies in teaching Object-Oriented Analysis and Design (OOAD). Experimental results indicate that the solution saves over 90% of drafting time, creates a diverse and realistic repository of learning materials, and effectively supports instructors in preventing student plagiarism through flexible scenario customization.

Keywords: Prompt Engineering, OOAD Case Study, Generative AI in Education.

Nhận bài: 13/01/2026

Phản biện: 09/02/2026

Duyệt đăng: 13/02/2026

I. ĐẶT VẤN ĐỀ

Trong chương trình đào tạo ngành Công nghệ thông tin (CNTT), học phần Phân tích thiết kế hệ thống thông tin hướng đối tượng (Object-Oriented Analysis and Design - OOAD) đóng vai trò then chốt, được coi là cầu nối giữa tư duy lập trình căn bản và năng lực kiến trúc hệ thống chuyên nghiệp. Khác với các môn lập trình thực thi thường tập trung vào cú pháp và giải thuật, OOAD đòi hỏi người học phải có khả năng trừu tượng hóa các bài toán thực tế phức tạp thành các mô hình đối tượng tinh gọn, thiết lập các mối quan hệ tương tác giữa chúng và đảm bảo các nguyên lý thiết kế bền vững như SOLID hay GRASP. Việc nắm vững OOAD giúp sinh viên không chỉ viết được mã nguồn chạy đúng, mà còn tạo ra được các hệ thống có khả năng bảo trì cao, dễ dàng mở rộng và linh hoạt trước các thay đổi về yêu cầu nghiệp vụ trong tương lai.

Công cụ quan trọng nhất và mang tính trực quan nhất để giảng dạy OOAD chính là các bài tập tình huống (Case Studies). Một Case Study chất lượng không chỉ dừng lại ở việc cung cấp bối cảnh nghiệp vụ thô, mà phải được tinh chỉnh để lồng ghép các thách thức về mặt logic. Điều này buộc sinh viên phải vận dụng tổng hợp các kỹ năng từ việc xác định tác nhân (Actors), xây dựng biểu đồ Use Case, cho đến việc chi tiết hóa cấu trúc bằng biểu đồ lớp (Class Diagram) và mô hình hóa hành vi thông qua các biểu đồ tương tác. Qua đó, sinh viên học được cách tối ưu hóa cấu trúc hệ

thống ngay từ giai đoạn thiết kế, giảm thiểu rủi ro sai sót trong giai đoạn triển khai thực tế.

II. NỘI DUNG NGHIÊN CỨU

2.1. Cơ sở lý thuyết

2.1.1. Thực trạng và thách thức

Mặc dù đóng vai trò nền tảng, việc xây dựng và ứng dụng Case Study trong giảng dạy OOAD hiện nay đang đối mặt với ba thách thức lớn mang tính hệ thống:

Sự khan hiếm và lỗi thời của học liệu: Các bài toán kinh điển như "Quản lý thư viện" hay "Quản lý ngân hàng" đã xuất hiện từ nhiều thập kỷ trước và xuất hiện dày đặc trong các giáo trình cũ. Những ví dụ này thường quá đơn giản và không còn phản ánh được hơi thở cũng như độ phức tạp của các hệ thống công nghệ hiện đại như mạng lưới Internet vạn vật (IoT), nền tảng chuỗi khối (Blockchain) hay các hệ thống tích hợp trí tuệ nhân tạo. Sự thiếu hụt các bối cảnh hiện đại khiến sinh viên khó hình dung được giá trị của thiết kế hướng đối tượng trong các dự án thực tế, từ đó dẫn đến sự sụt giảm đáng kể về hứng thú và động lực học tập.

Vấn nạn sao chép và sự hỗ trợ tiêu cực từ AI: Với sự phổ biến của các mô hình ngôn ngữ lớn (LLM), sinh viên dễ dàng tìm thấy lời giải chi tiết cho các bài toán kinh điển trên Internet hoặc yêu cầu các công cụ AI giải quyết toàn bộ bài tập chỉ trong vài giây. Khi các đề bài trở nên quá quen thuộc, AI có thể đưa ra đáp án chính xác mà không

cần người học phải tư duy. Hệ quả là tính khách quan trong việc đánh giá năng lực bị xâm phạm, đồng thời triệt tiêu khả năng phân biệt và động lực tư duy độc lập - vốn là yếu tố cốt lõi của một kiến trúc sư hệ thống tương lai.

Áp lực quá tải đối với đội ngũ giảng viên: Để thiết kế được một Case Study mới, có tính thực tế cao và độ khó được phân tầng phù hợp, giảng viên phải đầu tư rất nhiều thời gian để nghiên cứu kiến thức miền (Domain Knowledge) của lĩnh vực đó. Việc xây dựng các ràng buộc nghiệp vụ chặt chẽ, đảm bảo không có mâu thuẫn logic trong đề bài là một công việc đòi hỏi sự tỉ mỉ và sáng tạo lớn. Trong bối cảnh khối lượng giảng dạy và nghiên cứu khoa học ngày càng tăng, việc đổi mới liên tục kho học liệu để chống sao chép và bắt kịp xu hướng công nghệ trở thành một thách thức vượt quá khả năng về quỹ thời gian của nhiều giảng viên.

2.1.2. Cơ sở Khoa học và Công nghệ đề xuất

Sự bùng nổ của Trí tuệ nhân tạo tạo sinh (Generative AI), tiêu biểu là các mô hình ngôn ngữ lớn như GPT-4 hay Claude 3, đã mở ra những tiền đề quan trọng cho việc đổi mới phương pháp sáng tạo nội dung giáo dục.

Mô hình ngôn ngữ lớn (LLM): Các mô hình này sở hữu khả năng xử lý và sinh văn bản tự nhiên vượt trội dựa trên kho dữ liệu tri thức khổng lồ. Đặc biệt, LLM thể hiện sự am hiểu sâu sắc về các mô hình thiết kế phần mềm, cấu trúc dữ liệu và các phương pháp luận phát triển hệ thống hiện đại, cho phép chúng đóng vai trò như một cộng sự đắc lực trong việc phác thảo các kịch bản nghiệp vụ phức tạp. Các nghiên cứu gần đây đã chỉ ra năng lực của LLM trong việc hỗ trợ giảng viên xây dựng học liệu kỹ thuật một cách linh hoạt.

Kỹ thuật Prompt Engineering: Đây được xem là chiếc chìa khóa để khai thác tối đa tiềm năng của LLM. Bằng cách thiết kế các câu lệnh có cấu trúc, giảng viên có thể kiểm soát chặt chẽ đầu ra của AI. Các kỹ thuật như Role-playing giúp định hình phong cách phản hồi của AI như một chuyên gia phân tích nghiệp vụ, trong khi kỹ thuật Chain-of-Thought dẫn dắt AI đi qua từng bước suy luận logic để đảm bảo các Case Study sinh ra không bị hỏng về mặt nghiệp vụ. Việc ứng dụng đúng kỹ thuật câu lệnh sẽ giúp chuyển đổi các yêu cầu trừu tượng thành những bài tập tình huống có tính sư phạm cao và đúng định dạng yêu cầu của môn học.

2.1.3. Mục tiêu nghiên cứu

Bài báo này tập trung nghiên cứu và đề xuất một phương pháp luận cụ thể trong việc ứng dụng kỹ thuật Prompt Engineering để xây dựng quy

trình tự động hóa thiết kế Case Study đa lĩnh vực. Nghiên cứu hướng tới việc thiết lập một "Trợ lý ảo" thông minh cho giảng viên, có khả năng sản sinh ra các kịch bản nghiệp vụ độc nhất, sát với thực tế doanh nghiệp và tuân thủ nghiêm ngặt các tiêu chuẩn của OOAD. Kết quả của nghiên cứu không chỉ giúp giảm tải áp lực cho giảng viên mà còn góp phần nâng cao chất lượng học liệu, tăng cường tính minh bạch trong kiểm tra đánh giá và cá nhân hóa lộ trình phát triển tư duy thiết kế cho từng đối tượng sinh viên.

2.2. Phân tích các thành phần cốt lõi của một Case Study OOAD

Để AI có thể tạo ra các bài tập tình huống đạt chuẩn sư phạm và kỹ thuật, nghiên cứu xác định các thành phần dữ liệu trọng tâm mà một Case Study OOAD phải đáp ứng đầy đủ các yếu tố:

Bối cảnh nghiệp vụ: Cung cấp một tình huống thực tế cụ thể, chẳng hạn như Hệ thống quản lý trạm sạc xe điện thông minh hoặc Nền tảng đấu giá tài sản trực tuyến. Việc đa dạng hóa lĩnh vực giúp sinh viên không chỉ làm quen với nhiều thuật ngữ chuyên ngành mà còn hiểu được cách thức công nghệ giải quyết các bài toán kinh doanh đặc thù. Một bối cảnh được mô tả chi tiết sẽ giúp sinh viên rèn luyện kỹ năng phân tích thực tế, thay vì chỉ làm việc trên các mô hình lý thuyết khô khan.

Tác nhân và mục tiêu: Xác định rõ các đối tượng tương tác với hệ thống (Người dùng, Quản trị viên, Hệ thống bên thứ ba) và mục đích cụ thể của từng hành động.

Ví dụ: Trong hệ thống Nhà hàng, "Đầu bếp" không chỉ đơn thuần là người xem thực đơn, mà còn đóng vai trò là một Actor cập nhật trạng thái chế biến (đang chế biến, sẵn sàng) để bộ phận "Bồi bàn" có thể theo dõi thời gian thực. Việc xác định sai tác nhân sẽ dẫn đến việc thiết kế sai các sơ đồ tương tác sau này.

Quy trình nghiệp vụ và ngoại lệ: Mô tả chi tiết các luồng nghiệp vụ chuẩn và các tình huống phát sinh ngoài dự kiến.

Ví dụ: Quy trình thanh toán không chỉ dừng lại ở việc trừ tiền, mà phải mô tả rõ hệ thống sẽ xử lý thế nào nếu thẻ tín dụng bị từ chối, số dư không đủ, hoặc mã giảm giá đã hết hạn sử dụng. Các ngoại lệ này buộc sinh viên phải tư duy về tính bền vững và khả năng chịu lỗi của thiết kế hệ thống.

Ràng buộc thiết kế (Design Constraints): Đây là thành phần quan trọng nhất để định hướng tư duy OOAD của sinh viên. Giảng viên cần chủ

động lồng ghép các yêu cầu kỹ thuật bắt buộc vào đề bài, ví dụ: "Phải sử dụng tính đa hình để xử lý linh hoạt các loại hình thanh toán khác nhau" hoặc "Áp dụng Interface để tách biệt hoàn toàn logic lưu trữ dữ liệu với logic xử lý nghiệp vụ". Những ràng buộc này đóng vai trò là "kim chỉ nam" giúp sinh viên không đi chệch khỏi các nguyên lý thiết kế bền vững.

2.3. Quy trình thiết kế câu lệnh chuyên sâu

Nghiên cứu đề xuất cấu trúc "Master Prompt" dựa trên sự kết hợp tinh vi của các kỹ thuật tiên tiến nhằm kiểm soát chất lượng và tính nhất quán của đầu ra:

Kỹ thuật Role-playing (Đóng vai): Thiết lập một vị thế chuyên gia cho AI để định hình phong cách phản hồi.

Ví dụ lệnh: "Bạn là một Chuyên gia Phân tích Hệ thống (System Architect) có 20 năm kinh nghiệm triển khai các dự án ERP quy mô lớn và đồng thời là một Giảng viên ưu tú chuyên ngành OOAD. Hãy sử dụng văn phong kỹ thuật chuyên nghiệp nhưng có tính sư phạm cao để soạn thảo bài tập." Việc đóng vai giúp AI đưa ra các thuật ngữ chuẩn xác và cấu trúc đề bài mạch lạc hơn.

Kỹ thuật Few-shot Prompting: Cung cấp các mẫu bài tập chất lượng cao để AI học tập cấu trúc và định dạng mong muốn.

Ví dụ: Giảng viên đưa vào một bài tập mẫu về "Quản lý thư viện" đã được chuẩn hóa về mặt cấu trúc để AI hiểu rằng mỗi Case Study mới cần phải bao gồm đầy đủ các mục: Bối cảnh, Yêu cầu nghiệp vụ, Ràng buộc kỹ thuật và Câu hỏi thảo luận. Điều này giúp giảm thiểu việc AI sinh ra các nội dung rời rạc hoặc thiếu hụt thành phần quan trọng.

Kỹ thuật Chain-of-Thought: Yêu cầu AI thực hiện các bước suy luận trung gian thay vì đưa ra kết quả ngay lập tức.

Ví dụ lệnh: "Trước khi viết đề bài chi tiết, hãy liệt kê ra 5 thực thể chính của hệ thống này, xác định thuộc tính và các mối quan hệ giữa chúng (như Association, Aggregation, hay Composition). Sau khi đã nhất quán về cấu trúc dữ liệu, sau đó mới bắt đầu xây dựng câu chuyện nghiệp vụ xung quanh chúng." Kỹ thuật này đảm bảo tính logic xuyên suốt từ mô hình dữ liệu đến mô tả kịch bản.

2.4. Cấu trúc khung giải pháp đề xuất: Mô hình 3 giai đoạn

Quy trình tương tác giữa giảng viên và trợ lý AI được chuẩn hóa qua 3 giai đoạn chặt chẽ:

Giai đoạn 1: Xác định tham số đầu vào (Input Parameters)

Giảng viên đóng vai trò là người điều phối thông qua việc thiết lập các biến số đầu vào:

- Lĩnh vực ứng dụng: Hệ thống quản lý rạp chiếu phim thông minh tích hợp đặt vé qua ứng dụng di động.

- Trọng tâm kỹ thuật: Phân tích sự phụ thuộc giữa Suất chiếu, Phòng chiếu và Phim. Yêu cầu sử dụng Observer Pattern.

- Mức độ khó: Nâng cao.

Giai đoạn 2: Thực thi Master Prompt và sinh nội dung

Ở giai đoạn này, hệ thống sẽ xử lý các tham số để tạo ra một đề bài hoàn chỉnh với các kịch bản tương tác phức tạp. AI sinh ra nội dung đáp ứng đầy đủ các thành phần đã phân tích ở trên.

Giai đoạn 3: Tinh chỉnh và cá nhân hóa

Giảng viên thực hiện các điều chỉnh dựa trên phản hồi thực tế:

Lệnh điều chỉnh: "Hãy bổ sung thêm kịch bản về quản lý khách hàng thân thiết để sinh viên phải áp dụng nguyên lý Open/Closed Principle (OCP)."

2.5. Tiêu chí đánh giá chất lượng (Rubric)

Để đảm bảo học liệu có giá trị giáo dục cao, nghiên cứu đề xuất bộ tiêu chí kiểm định:

- Độ bao phủ kiến thức: Đảm bảo áp dụng được các nguyên lý OOAD.

- Tính phi cấu trúc: Đề bài đủ "mở" để sinh viên tự suy luận.

- Khả năng kiểm chứng nhanh: Hỗ trợ giảng viên tạo đáp án mẫu để đối chiếu logic.

2.6. Kết quả thực nghiệm và kết luận

2.6.1. Kết quả thực nghiệm tạo bộ học liệu đa lĩnh vực

Thông qua quy trình đề xuất, nghiên cứu đã tiến hành thực nghiệm sinh 15 Case Study thuộc các lĩnh vực khác nhau như Fintech, Agritech, Smart City và EdTech. Kết quả cho thấy 100% các đề bài được tạo ra đều đảm bảo được cấu trúc nghiệp vụ mạch lạc và lồng ghép thành công các thách thức thiết kế hướng đối tượng phức tạp. Sự đa dạng này giúp giảng viên nhanh chóng thay đổi học liệu theo từng học kỳ mà không gặp phải sự lặp lại nhàm chán của các bài toán truyền thống.

Đặc biệt, kỹ thuật Chain-of-Thought đã chứng minh được sức mạnh trong việc loại bỏ các lỗi mâu thuẫn logic về mối quan hệ giữa các đối tượng. Trong các thử nghiệm so sánh với câu lệnh thông thường, AI thường xuyên nhầm lẫn giữa mối quan hệ Aggregation (tập hợp) và Composition (cấu thành) hoặc xác định sai quan hệ "Is-a" và "Has-a". Tuy nhiên, với lộ trình tư duy bước-

theo-bước, các sai sót này đã giảm xuống mức tối thiểu, tạo ra những sơ đồ lớp mẫu có độ chính xác cao về mặt kiến trúc, giúp giảng viên yên tâm hơn khi sử dụng học liệu trực tiếp trên lớp.

2.6.2. *Đánh giá hiệu suất so với phương pháp truyền thống*

Công đoạn	Phương pháp Truyền thống (Phút)	Phương pháp Đề xuất (Phút)	Tỷ lệ tiết kiệm (%)
Nghiên cứu nghiệp vụ lĩnh vực mới	120	5	95.8
Soạn thảo kịch bản và ràng buộc	90	10	88.9
Thiết lập tiêu chí chấm & Đáp án mẫu	30	5	83.3
Tổng cộng	240	20	91.7

Số liệu thực tế cho thấy giải pháp ứng dụng Prompt Engineering không chỉ đơn thuần là công cụ hỗ trợ mà còn là một bước đột phá về năng suất lao động. Việc tiết kiệm hơn 90% thời gian cho phép giảng viên tập trung nhiều hơn vào các hoạt động sư phạm chuyên sâu như: tư vấn trực tiếp cho các nhóm sinh viên, tinh chỉnh các nút thắt logic trong bài toán để kích thích tư duy phản biện, thay vì mất quá nhiều công sức cho các công việc tra cứu nghiệp vụ thô sơ từ đầu.

2.6.3. *Chất lượng và tính sư phạm*

Qua đánh giá từ hội đồng chuyên môn, các Case Study được tạo ra đạt độ phức tạp cần thiết để đánh giá toàn diện năng lực của sinh viên. Tuy nhiên, thảo luận cũng chỉ ra những khía cạnh quan trọng để đảm bảo giá trị giáo dục:

Vai trò trung tâm của Giảng viên: AI đóng vai trò là "người tạo bản thảo", nhưng giảng viên phải là "người thẩm định" cuối cùng. Giảng viên cần kiểm soát để đảm bảo các thuật ngữ chuyên ngành phù hợp với văn hóa kinh doanh và các quy chuẩn pháp lý tại địa phương (Ví dụ: quy trình thanh toán trong Fintech phải tuân thủ quy định của Ngân hàng Nhà nước Việt Nam).

Cá nhân hóa và ngăn chặn sao chép: Một ưu điểm vượt trội là khả năng tạo ra các biến thể kịch bản. Giảng viên có thể yêu cầu AI tạo ra 10 biến thể khác nhau từ cùng một bối cảnh (ví dụ: cùng quản lý khách

Nghiên cứu tiến hành khảo sát thời gian thực hiện trên nhóm 05 giảng viên có kinh nghiệm giảng dạy bộ môn OOAD trên 5 năm. Kết quả ghi nhận sự chênh lệch rõ rệt về hiệu suất làm việc giữa quy trình thủ công và quy trình có hỗ trợ của AI:

sạn nhưng phiên bản A tập trung vào đặt phòng, phiên bản B tập trung vào quản lý kho vật tư) với cùng mức độ khó. Điều này cho phép cấp phát đề bài riêng biệt cho từng nhóm, triệt tiêu khả năng sao chép và buộc sinh viên phải tự thân vận động trong việc phân tích.

III. KẾT LUẬN

Nghiên cứu đã khẳng định tính hiệu quả và khả thi của kỹ thuật Prompt Engineering trong việc hiện đại hóa quy trình soạn thảo học liệu OOAD. Giải pháp không chỉ cung cấp những bài tập tình huống giàu tính thực tế, đa dạng mà còn tạo ra rào cản bảo mật vững chắc trước việc lạm dụng AI của người học, đồng thời giải phóng đáng kể thời gian sáng tạo cho đội ngũ giảng viên.

Trong tương lai, nghiên cứu đề xuất mở rộng theo các hướng sau:

Tích hợp Vision AI: Tự động hóa việc chấm điểm sơ đồ UML từ ảnh chụp hoặc file vẽ của sinh viên, cung cấp phản hồi tức thời.

Cá nhân hóa độ khó: Tự động điều chỉnh độ phức tạp của bài tập dựa trên dữ liệu hiệu suất của từng sinh viên trong hệ thống quản lý học tập (LMS).

Mở rộng hệ sinh thái liên môn: Áp dụng mô hình Master Prompt cho các môn học như Thiết kế Cơ sở dữ liệu và Kiểm thử phần mềm để tạo ra một chuỗi đề án nhất quán, giúp sinh viên phát triển một dự án xuyên suốt nhiều học kỳ.

TÀI LIỆU THAM KHẢO

- Mollick, E. R., & Mollick, L. (2023). *Using AI to Implement Error-Free Scaffolding: A Guide for Educators*. Wharton School Press.
- White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., & Schmidt, D. C. (2023). *A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT*. arXiv preprint arXiv:2302.11382.
- Baidoo-Anu, D., & Ansah, L. O. (2023). *Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning*. Journal of AI.
- Becker, B. A., et al. (2023). *Programming Is Hard - Or at Least It Used to Be: Educational Opportunities and Challenges of AI Code Generation*. In Proceedings of the 54th ACM Technical Symposium on Computer Science Education (SIGCSE).
- Zhou, J., et al. (2023). *Learn Prompting: A Comprehensive Guide to Prompt Engineering*. Open-source educational resource.