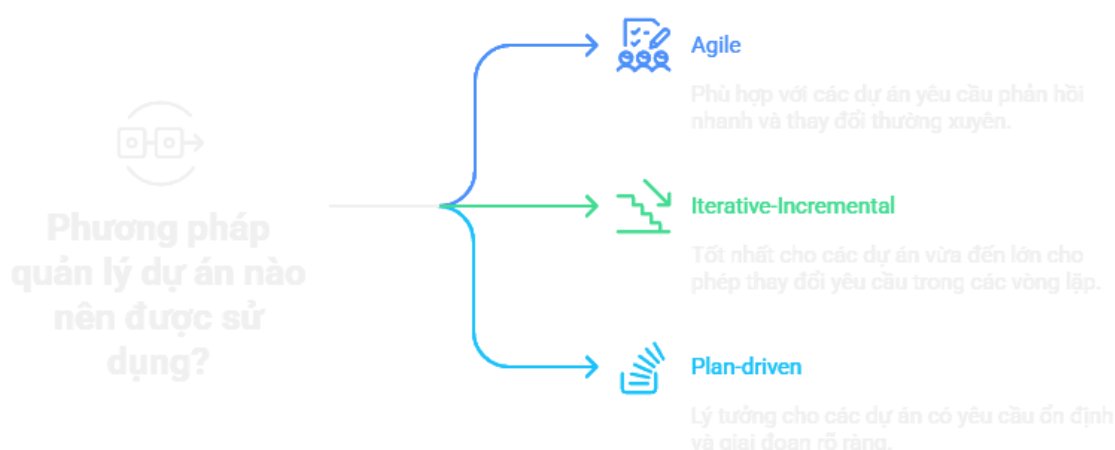
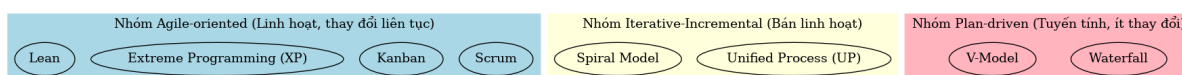


Methodology

■ Section	
■ Status 1	Not started



Made with Napkin

1. Nhóm linh hoạt, thay đổi liên tục (Agile-oriented)

- **Scrum**Ref Scrum v2
- **Kanban**Ref Kanban v2
- **XP (Extreme Programming)**Ref XP v2
- **Lean**Ref Lean v2

◆ Điểm chung:

- Yêu cầu có thể thay đổi thường xuyên.
- Phát triển theo vòng lặp/ngắn hạn.
- Người dùng tham gia cao.

- Phù hợp môi trường cần phản hồi nhanh.

2. Nhóm bán linh hoạt, lặp lại có cấu trúc (Iterative-Incremental)

- **Unified Process (UP)**Ref Unified Process v2
- **Spiral Model**Ref Spiral Model v2

◆ Điểm chung:

- Phát triển qua nhiều vòng lặp, có giai đoạn rõ ràng.
- Cho phép thay đổi yêu cầu ở giữa các vòng.
- Yêu cầu ban đầu ở mức cao, chi tiết hóa dần.
- Phù hợp dự án vừa đến lớn, có rủi ro.

3. Nhóm tuyến tính, ít thay đổi (Plan-driven)

- **Waterfall**Ref Waterfall v2
- **V-Model**Ref V model v2Ref V model v2

◆ Điểm chung:

- Yêu cầu ổn định, xác định ngay từ đầu.
- Không linh hoạt khi thay đổi.
- Có giai đoạn phát triển rõ ràng, thường áp dụng kiểm thử sau khi hoàn tất từng giai đoạn (V-Model gắn kiểm thử song song).

Nếu xét mức độ giống nhau nhất:

- **Scrum, Kanban, XP, Lean** gần nhau (đều thuộc Agile).
- **Unified Process và Spiral** gần nhau (iterative + risk/feedback).
- **Waterfall và V-Model** gần nhau (plan-driven, yêu cầu cố định).

TEMPLATE 1 – Agile-oriented (Scrum / Kanban / XP / Lean)

1. General Introduction

The project [...] involves requirements that change frequently and are difficult to fully define at the outset. The development team needs a flexible method that can respond quickly to evolving needs from [...] and the market. Therefore, we choose the [...] model, which belongs to the Agile-oriented group.

2. Adaptability to Change

- Agile allows new requirements to be added or adjusted at any time without being restricted by an upfront fixed plan.
- The work process is divided into short iterations/sprints ([...]), ensuring that after each cycle, the team can review, adapt, and enhance the product based on real feedback.
- This is particularly suitable when project success depends on trends or user behaviors that can change rapidly.

3. User Involvement

- Users or their representatives ([...]) are continuously involved throughout the development process — from defining requirements to reviewing deliverables and adjusting priorities.
- This high involvement ensures the final product closely matches real-world needs and boosts user satisfaction.

4. Team Size and Characteristics

- This model suits teams of [...] members who are self-organizing, communicate directly, and collaborate closely.
- Members often take on multiple roles, reducing waiting time between tasks and increasing productivity.

5. Conclusion

Given its adaptability, high user involvement, and fit for the team structure, the [...] model will enable the [...] project to achieve its objectives in a dynamic environment.

TEMPLATE 2 – Iterative-Incremental (Unified Process / Spiral)

1. General Introduction

The project [...] has some requirements clearly defined, while others need to be explored and validated during execution. Therefore, we choose the [...] model from the Iterative-Incremental group to leverage the benefits of iterative cycles and progressive elaboration of requirements.

2. Iterative Development Structure

- The process is broken down into multiple iterations, each including analysis, design, development, testing, and review.
- Every iteration produces a more refined version of the product than the last, enabling early trials and quick feedback collection.
- This helps reduce risks by identifying and resolving issues early.

3. Effective Risk Management

- The [...] model places strong emphasis on risk analysis in every cycle (**Spiral**) or manages requirements via detailed use cases (**Unified Process**).
- This ensures potential issues are addressed before large-scale deployment or costly changes.

4. Continuous Improvement and Adaptation

- After each iteration, the team can adapt requirements, design, or implementation based on user feedback from [...].
- This improves the alignment of the product with evolving business or technical needs.

5. Team Size

- Suitable for medium- to large-sized teams capable of splitting roles across phases or functions.
- Multiple iterations allow parallel work streams, which shortens total development time.

6. Conclusion

With its balance between structure and flexibility, the [...] model ensures that the [...] project can achieve high quality, minimize risks, and respond effectively to changing requirements during development.

TEMPLATE 3 – Plan-driven (Waterfall / V-Model)

1. General Introduction

The project [...] has stable, well-defined requirements that are unlikely to change during development. The team seeks a structured model to control schedule and cost effectively. Therefore, we choose the [...] model from the Plan-driven group.

2. Early and Clear Requirement Definition

- All major requirements are gathered and documented in detail from the start.
- This fits perfectly with the Plan-driven approach, allowing a comprehensive project plan to be established and reducing unexpected scope changes.

3. Structured Development Process

- The process is divided into clearly defined phases ([...]), each producing specific deliverables before proceeding to the next.
- This facilitates easier monitoring of progress, quality, and resource allocation.

4. Testing Strategy

- In Waterfall, testing is performed after the development phase is complete.
- In the V-Model, each development phase is paired with a corresponding testing phase, enabling earlier defect detection.

5. Team Size and User Involvement

- Works best with medium to large teams with clearly separated roles (analysts, designers, developers, testers).
- User involvement is concentrated in the requirements and acceptance phases, ensuring focus during implementation.

6. Conclusion

With stable requirements and a high degree of process control, the [...] model is well-suited to delivering the [...] project on time, with consistent quality and predictable costs.

MẪU 1 – Agile-oriented (Scrum / Kanban / XP / Lean)

1. Giới thiệu chung về lý do lựa chọn

Dự án [...] có đặc điểm yêu cầu thay đổi thường xuyên, khó xác định đầy đủ ngay từ giai đoạn đầu. Nhóm phát triển cần một phương pháp linh hoạt, có khả năng phản hồi nhanh với những thay đổi từ [...] và thị trường. Vì vậy, chúng tôi lựa chọn mô hình [...] thuộc nhóm Agile-oriented.

2. Khả năng thích ứng với thay đổi

- Agile cho phép bổ sung hoặc điều chỉnh yêu cầu bất cứ lúc nào, không bị ràng buộc bởi kế hoạch cố định ban đầu.
- Quy trình làm việc được chia thành các vòng lặp/ngắn hạn ([...]), đảm bảo sau mỗi vòng, nhóm có thể rà soát, điều chỉnh và bổ sung tính năng dựa trên phản hồi thực tế.
- Điều này đặc biệt phù hợp khi yêu cầu của dự án phụ thuộc nhiều vào xu hướng hoặc hành vi người dùng, vốn có thể thay đổi nhanh chóng.

3. Sự tham gia của người dùng

- Người dùng hoặc đại diện của họ ([...]) sẽ liên tục tham gia vào quá trình phát triển, từ việc xác định yêu cầu, phản hồi kết quả đến điều chỉnh ưu tiên.

- Điều này giúp sản phẩm cuối cùng bám sát nhu cầu thực tế và nâng cao mức độ hài lòng.

4. Quy mô và đặc điểm nhóm phát triển

- Mô hình phù hợp với nhóm [...] thành viên, có khả năng tự tổ chức, giao tiếp trực tiếp và phối hợp chặt chẽ.
- Các thành viên đều có thể đảm nhiệm nhiều vai trò, giúp giảm thời gian chờ đợi và tăng hiệu quả.

5. Kết luận

Với các ưu điểm về khả năng thích ứng, sự tham gia của người dùng và quy mô nhóm phù hợp, mô hình [...] sẽ giúp dự án [...] đạt được mục tiêu trong môi trường nhiều biến động.

MẪU 2 – Iterative-Incremental (Unified Process / Spiral)

1. Giới thiệu chung về lý do lựa chọn

Dự án [...] có một phần yêu cầu đã được xác định rõ ràng, nhưng vẫn còn nhiều hạng mục cần làm rõ và kiểm chứng trong quá trình thực hiện. Vì vậy, chúng tôi lựa chọn mô hình [...] thuộc nhóm Iterative-Incremental, nhằm tận dụng lợi thế của việc phát triển theo vòng lặp và bổ sung dần yêu cầu.

2. Cấu trúc phát triển theo vòng lặp

- Quy trình được chia thành nhiều vòng lặp (iteration), mỗi vòng bao gồm các bước: phân tích, thiết kế, phát triển, kiểm thử và đánh giá kết quả.
- Mỗi vòng lặp giúp dự án tạo ra một phiên bản hoàn thiện hơn so với vòng trước, có thể triển khai thử nghiệm và lấy phản hồi ngay.
- Điều này giúp giảm rủi ro vì các vấn đề được phát hiện và xử lý sớm.

3. Quản lý rủi ro hiệu quả

- Mô hình [...] đặc biệt chú trọng đến việc phân tích rủi ro ở mỗi vòng lặp (nếu là Spiral), hoặc quản lý yêu cầu thông qua các use case (nếu là UP).
- Cách tiếp cận này đảm bảo rằng trước khi mở rộng tính năng hoặc triển khai trên diện rộng, các rủi ro tiềm ẩn đã được đánh giá và xử lý.

4. Khả năng thích ứng và cải tiến liên tục

- Sau mỗi vòng lặp, nhóm phát triển có thể điều chỉnh yêu cầu, thiết kế hoặc cách triển khai dựa trên phản hồi của [...].
- Điều này giúp sản phẩm đáp ứng đúng nhu cầu thực tế hơn so với việc cố định toàn bộ yêu cầu ngay từ đầu.

5. Quy mô nhóm phát triển

- Mô hình phù hợp với nhóm vừa đến lớn, có khả năng phân chia vai trò theo giai đoạn hoặc chức năng.
- Việc có nhiều vòng lặp giúp các nhóm làm việc song song, rút ngắn tổng thời gian phát triển.

6. Kết luận

Nhờ cấu trúc linh hoạt nhưng vẫn có tính tổ chức, mô hình [...] sẽ đảm bảo dự án [...] vừa đạt được chất lượng, vừa giảm thiểu rủi ro, vừa đáp ứng các yêu cầu thay đổi trong quá trình thực hiện.

MẪU 3 – Plan-driven (Waterfall / V-Model)

1. Giới thiệu chung về lý do lựa chọn

Dự án [...] có yêu cầu rõ ràng, ổn định và ít có khả năng thay đổi trong suốt quá trình thực hiện. Đội ngũ phát triển mong muốn một mô hình có cấu trúc chặt chẽ, dễ quản lý tiến độ và chi phí. Vì vậy, chúng tôi lựa chọn mô hình [...] thuộc nhóm Plan-driven.

2. Xác định yêu cầu sớm và rõ ràng

- Ngay từ đầu, tất cả yêu cầu quan trọng đã được thu thập và tài liệu hóa một cách chi tiết.
- Điều này phù hợp với đặc điểm của mô hình Plan-driven, giúp xây dựng kế hoạch tổng thể và hạn chế phát sinh không mong muốn trong quá trình triển khai.

3. Quy trình phát triển có cấu trúc

- Quy trình được chia thành các giai đoạn rõ ràng ([...]), mỗi giai đoạn có đầu ra cụ thể trước khi chuyển sang giai đoạn tiếp theo.

- Điều này giúp dễ dàng kiểm soát tiến độ, chất lượng và phân bổ nguồn lực.

4. Kiểm thử song song hoặc theo giai đoạn

- Nếu áp dụng Waterfall, kiểm thử sẽ được thực hiện sau khi hoàn thành phát triển.
- Nếu áp dụng V-Model, mỗi giai đoạn phát triển sẽ được gắn với một giai đoạn kiểm thử tương ứng, giúp phát hiện lỗi sớm hơn.

5. Quy mô nhóm và sự tham gia của người dùng

- Mô hình phù hợp với nhóm vừa đến lớn, có phân công vai trò rõ ràng (phân tích, thiết kế, lập trình, kiểm thử).
- Người dùng tham gia chủ yếu ở giai đoạn đầu (lấy yêu cầu) và giai đoạn cuối (thử nghiệm), giúp tập trung nguồn lực phát triển ở giữa dự án.

6. Kết luận

Với sự ổn định của yêu cầu và tính tổ chức cao, mô hình [...] sẽ giúp dự án [...] hoàn thành đúng tiến độ, đảm bảo chất lượng và tiết kiệm chi phí.

#learn
toknow

KHÓA HỌC CẤP TỐC

Uy tín chất lượng
giá mềm



Kỳ 1.

MAE + PRF

Kỳ 2.

PRO + MAD

Kỳ 3.

LAB + CSD + DBI
+ WED201c + JPD113

Kỳ 4.

PRJ + MAS + JPD + SWE

Kỳ 5.

FER + SWP391 + ISP392



HƯNG HƯNG



@FORUSER2761



4USER.NET