Mô hình tái sử dụng

* Yêu cầu: - ( Vì trong mô hình này, yêu cầu sẽ được điều chỉnh trong khi kĩ sư phần mềm đặc tả yêu cầu và phân tích thành phần sẵn có)
* Tương tác khách hàng: - (Trong mô hình này, kĩ sư phần mềm sẽ dựa
* Quy mô dự án: - (Ý kiến cá nhân: việc sử dụng mô hình này, phần mềm có thể sẽ khá lỗi thời do dùng lại tài nguyên có sẵn cùng với đó là sẽ không có nhiều tài nguyên có sẵn để sử dụng khiến cho kĩ sư phần mềm không thể đi đường dài để theo đuổi 1 dự án lớn)
* Ưu điểm:

+ Phát triển một cách có hệ thống các thành phần tái sử dụng

+ Tái sử dụng một cách có hệ thống các thành phần này như các khối được xây dựng để tạo hệ thống mới

+ Một thành phần tái sử dụng có có thể là mã, nhưng ích lợi lớn hơn của việc tái sử dụng là tầm nhìn cao và rộng hơn về những gì có thể tái sử dụng

* Nhược điểm:
  + Năng lực tái sử dụng hạn chế (Limited Reuse Potential)

+ Hạn chế khả năng tìm kiếm một tài nguyên tương thích.

* Tổng phí đáng kể.

+ Tổng phí thực chất đã bị trừ đi phí từ ích lợi của thành phần tái sử dụng.

* + Trở ngại đáng kể (Non-negligible Obstacles)

+ Thay đổi đáng kể các thủ tục thao tác.

+ Chỉ đầu tư ngắn hạn, nhưng muốn đạt lợi ích dài hạn.

Mô hình tiến hóa

* Yêu cầu: - (Vì trong mô hình này, kĩ sư phần mềm sẽ chỉnh mẫu thử qua nhiều phiên bản để phù hợp với yêu cầu của khách hàng)
* Tương tác khách hàng: + (Việc điều chỉnh nhiều lần sẽ buộc kĩ sư phải tương tác với khách hàng nhiều hơn để làm ra sản phẩm cuối cùng)
* Quy mô dự án: + ( Ý kiến cá nhân: việc có nhiều mẫu thử để phù hợp yêu cầu khách hàng khiến cho mô hình này khá phù hợp với những dự án lớn)
* Ưu điểm:

+ Phù hợp với dự án vừa và nhỏ.

+ Phù hợp với các phần của dự án phức tạp.

+ Phù hợp với các hệ thống có vòng đời ngắn.

* Nhược điểm:

+ Cấu trúc hệ thống tồi.

+ Tiến trình không rõ ràng.