Chương 9: Lập kế hoạch và ước tính.

Mục tiêu học tập

Sau khi nghiên cứu chương này, bạn sẽ có thể

• Giải thích tầm quan trọng của việc lập kế hoạch.

• Ước tính quy mô và chi phí xây dựng một sản phẩm phần mềm.

• Đánh giá cao tầm quan trọng của việc cập nhật và theo dõi các ước tính.

• Lập kế hoạch quản lý dự án phù hợp với tiêu chuẩn IEEE.

Những thách thức khi xây dựng một sản phẩm phần mềm không có giải pháp dễ dàng. Để kết hợp một sản phẩm phần mềm lớn với nhau cần thời gian và tài nguyên. Và, giống như bất kỳ công trình xây dựng lớn dự án nào khác, lập kế hoạch cẩn thận khi bắt đầu dự án có lẽ là yếu tố quan trọng nhất giúp phân biệt thành công và thất bại. Kế hoạch ban đầu này, tuy nhiên, không phương tiện là đủ. Lập kế hoạch, giống như kiểm thử, phải tiếp tục trong suốt quá trình phát triển và bảo trì phần mềm. Bất chấp nhu cầu lập kế hoạch liên tục, những các hoạt động đạt đến đỉnh điểm sau khi các thông số kỹ thuật đã được vạch ra nhưng trước khi các hoạt động thiết kế bắt đầu. Tại thời điểm này trong quá trình, khoảng thời gian có ý nghĩa và ước tính chi phí là được tính toán và lập kế hoạch chi tiết để hoàn thành dự án.

Trong chương này, chúng tôi phân biệt hai loại quy hoạch này, quy hoạch tiến hành trong suốt dự án và việc lập kế hoạch căng thẳng phải được thực hiện khi các thông số kỹ thuật đã hoàn thành.

9.1 Lập kế hoạch và Quy trình Phần mềm

Tốt nhất, chúng tôi muốn lập kế hoạch cho toàn bộ dự án phần mềm ngay từ đầu của quá trình, và sau đó thực hiện theo kế hoạch đó cho đến khi phần mềm đích cuối cùng đã được chuyển giao cho khách hàng. Tuy nhiên, điều này là không thể, vì chúng tôi thiếu đủ thông tin trong quy trình làm việc ban đầu để có thể vạch ra một kế hoạch có ý nghĩa cho dự án hoàn chỉnh. Ví dụ, trong quá trình yêu cầu quy trình làm việc, bất kỳ loại lập kế hoạch nào (ngoại trừ chỉ cho chính quy trình công việc yêu cầu) đều vô ích.

Có một thế giới khác biệt giữa thông tin do nhà phát triển sử dụng tại kết thúc quy trình công việc yêu cầu và vào cuối quy trình công việc phân tích, tương tự như sự khác biệt giữa bản phác thảo thô và bản thiết kế chi tiết. Theo yêu cầu cuối cùng quy trình làm việc, các nhà phát triển tốt nhất phải có hiểu biết không chính thức về những gì khách hàng cần.

Ngược lại, vào cuối luồng công việc phân tích, tại thời điểm đó khách hàng ký một tài liệu nêu chính xác những gì sẽ được xây dựng, các nhà phát triển đánh giá chi tiết về hầu hết (nhưng thường vẫn không phải là tất cả) các khía cạnh của sản phẩm mục tiêu. Đây là điểm sớm nhất trong tại đó có thể xác định chính xác khoảng thời gian và ước tính chi phí.

Tuy nhiên, trong một số tình huống, một tổ chức có thể được yêu cầu đưa ra thời hạn

và ước tính chi phí trước khi các trích dẫn cụ thể có thể được lập. Trong trường hợp xấu nhất, khách hàng có thể

nhấn mạnh vào một giá thầu trên cơ sở một hoặc hai giờ thảo luận sơ bộ. Hình 9.1 cho thấy

làm thế nào vấn đề này có thể được. Dựa trên một mô hình trong [Boehm và cộng sự, 2000], nó mô tả phạm vi ước tính chi phí tương đối cho các quy trình công việc khác nhau của vòng đời. Ví dụ, giả sử

rằng, khi một sản phẩm vượt qua kiểm tra chấp nhận ở cuối quy trình triển khai

và được giao cho khách hàng, giá của nó là 1 triệu đô la. Nếu một ước tính chi phí có

được thực hiện giữa chừng với quy trình công việc yêu cầu, có khả năng là nó sẽ được

ở đâu đó trong phạm vi (0,25 triệu đô la, 4 triệu đô la), như thể hiện trong Hình 9.2. Tương tự, nếu

ước tính chi phí đã được thực hiện giữa chừng trong quá trình phân tích, phạm vi khả năng

ước tính sẽ giảm xuống (0,5 triệu đô la, 2 triệu đô la). Hơn nữa, nếu ước tính chi phí

đã được thực hiện vào cuối quy trình phân tích, nghĩa là, vào thời điểm thích hợp, kết quả

có lẽ sẽ vẫn nằm trong phạm vi tương đối rộng (0,67 triệu đô la, 1,5 triệu đô la).

Tất cả bốn điểm được đánh dấu trên các đường giới hạn trên và dưới trong Hình 9.2, có

thang logarit trên trục tung. Mô hình này được gọi là hình nón của sự không chắc chắn.

|  |
| --- |
|  |

Rõ ràng từ Hình 9.1 và 9.2 rằng ước tính chi phí không phải là một khoa học chính xác; lý do cho điều này

được đưa ra trong Phần 9.2.

Dữ liệu mà mô hình hình nón của sự không chắc chắn dựa trên đó là cũ, bao gồm năm đề xuất được đệ trình lên Bộ phận Hệ thống Điện tử của Lực lượng Không quân Hoa Kỳ [Devenny, 1976], và

kỹ thuật ước lượng đã được cải thiện kể từ thời điểm đó. Tuy nhiên, hình dạng tổng thể của

đường cong trong Hình 9.1 có thể không thay đổi quá mức. Do đó, thời hạn hoặc ước tính chi phí quá sớm, tức là ước tính được thực hiện trước khi các báo cáo cụ thể được ký kết

do khách hàng bật ra, có thể kém chính xác hơn đáng kể so với ước tính được đưa ra khi

dữ liệu đầy đủ đã được tích lũy.

Bây giờ chúng tôi kiểm tra các kỹ thuật để ước tính thời lượng và chi phí. Giả định trong suốt phần còn lại của chương này là quy trình phân tích đã được hoàn thành; đó là,

ước tính và lập kế hoạch có ý nghĩa bây giờ có thể được thực hiện.

9.2 Thời lượng và chi phí ước tính

Ngân sách là một phần không thể thiếu của bất kỳ kế hoạch quản lý dự án phần mềm nào. Trước khi thiết kế

bắt đầu, khách hàng cần biết họ sẽ phải trả bao nhiêu cho sản phẩm.

Nếu nhóm phát triển đánh giá thấp chi phí thực tế, tổ chức phát triển có thể

mất tiền vào dự án. Mặt khác, nếu nhóm phát triển đánh giá quá cao, thì

khách hàng có thể quyết định rằng, trên cơ sở phân tích chi phí - lợi ích hoặc lợi tức đầu tư,

không có ích gì khi sản phẩm được xây dựng. Ngoài ra, khách hàng có thể giao công việc cho

một tổ chức phát triển khác có ước tính hợp lý hơn. Dù bằng cách nào, nó là rõ ràng rằng ước tính chi phí chính xác là rất quan trọng.

Trên thực tế, có hai loại chi phí liên quan đến việc phát triển phần mềm. Điều đầu tiên là chi phí nội bộ, chi phí cho các nhà phát triển; thứ hai là chi phí bên ngoài, giá mà khách hàng sẽ trả tiền. Chi phí nội bộ bao gồm tiền lương của các nhóm phát triển, người quản lý và nhân viên hỗ trợ tham gia vào dự án; chi phí của phần cứng và phần mềm để phát triển sản phẩm; và chi phí chung như tiền thuê nhà, tiền điện nước và tiền lương của

quản lí cấp cao. Mặc dù giá thường dựa trên chi phí cộng với tỷ suất lợi nhuận,

trong một số trường hợp, các yếu tố kinh tế và tâm lý là quan trọng. Ví dụ, các nhà phát triển

những người thực sự cần công việc có thể được chuẩn bị để tính phí khách hàng bằng giá. Một khác

tình huống phát sinh khi hợp đồng được trao trên cơ sở hồ sơ dự thầu. Khách hàng có thể từ chối

giá thầu thực sự thấp hơn đáng kể so với tất cả các giá thầu khác vì lý do chất lượng của

kết quả sản phẩm có lẽ cũng sẽ thấp hơn đáng kể. Do đó, một nhóm phát triển

có thể cố gắng đưa ra một giá thầu sẽ thấp hơn một chút, nhưng không đáng kể, thấp hơn giá thầu

tin rằng sẽ là giá thầu của đối thủ cạnh tranh.

Một phần quan trọng khác của bất kỳ kế hoạch nào là ước tính thời gian của dự án. Khách hàng

chắc chắn muốn biết khi nào sản phẩm fi nished sẽ được giao. Nếu sự phát triển

tổ chức không thể giữ đúng lịch trình của mình, thì tốt nhất là tổ chức mất uy tín,

tại các điều khoản hình phạt tồi tệ nhất được viện dẫn. Trong mọi trường hợp, những người quản lý chịu trách nhiệm về phần mềm

kế hoạch quản lý dự án có rất nhiều việc phải làm. Ngược lại, nếu sự phát triển

tổ chức đánh giá quá cao thời gian cần thiết để xây dựng sản phẩm, khi đó sẽ có cơ hội tốt

rằng khách hàng sẽ đi nơi khác.

Thật không may, không có nghĩa là dễ dàng để có được một ước tính chi phí chính xác và thời gian

ước tính. Có quá nhiều biến liên quan để có thể xử lý chính xác một trong hai

chi phí hoặc thời lượng. Một khó khăn lớn là yếu tố con người. Hơn 40 năm trước, Sackman và

đồng nghiệp đã quan sát thấy sự khác biệt lên tới 28-1 giữa các cặp lập trình viên [Sackman,

Erikson, và Grant, 1968]. Thật dễ dàng để cố gắng phủ nhận kết quả của họ bằng cách nói rằng các lập trình viên có kinh nghiệm luôn làm tốt hơn những người mới bắt đầu, nhưng Sackman và các đồng nghiệp của ông đã so sánh

các cặp lập trình viên phù hợp. Ví dụ, họ quan sát thấy hai lập trình viên với 10 năm

kinh nghiệm về các loại dự án tương tự và đo lường thời gian chúng thực hiện

các tác vụ như mã hóa và gỡ lỗi. Sau đó, họ quan sát, chẳng hạn, hai người mới bắt đầu đã tham gia

nghề nghiệp trong cùng một khoảng thời gian ngắn và có trình độ học vấn tương tự. So sánh hiệu suất kém nhất và tốt nhất, họ quan sát thấy sự khác biệt từ 6 đến 1 về kích thước sản phẩm, 8 đến

1 trong thời gian thực thi sản phẩm, 9 đến 1 trong thời gian phát triển, 18 đến 1 trong thời gian mã hóa và 28 đến 1 trong

thời gian gỡ lỗi. Một quan sát đặc biệt đáng báo động là màn trình diễn tốt nhất và tồi tệ nhất

trên một sản phẩm của hai lập trình viên, mỗi người đã có 11 năm kinh nghiệm. Thậm chí

khi các trường hợp tốt nhất và xấu nhất được loại bỏ khỏi mẫu của Sackman và cộng sự, sự khác biệt quan sát được vẫn theo thứ tự từ 5 đến 1. Trên cơ sở các kết quả này, rõ ràng, chúng tôi không thể hy vọng

để ước tính chi phí hoặc thời lượng phần mềm với bất kỳ mức độ chính xác nào (trừ khi chúng tôi đã nêu chi tiết

thông tin liên quan đến tất cả các kỹ năng của tất cả các nhân viên, điều này sẽ là bất thường nhất). Nó

đã được lập luận rằng, trong một dự án lớn, sự khác biệt giữa các cá nhân có xu hướng triệt tiêu,

nhưng đây có lẽ là mơ tưởng; sự hiện diện của một hoặc hai đội rất tốt (hoặc rất tệ)

các thành viên có thể gây ra sự sai lệch rõ rệt so với lịch trình và ảnh hưởng đáng kể đến ngân sách.

Một yếu tố con người khác có thể ảnh hưởng đến ước tính là ở một quốc gia tự do, không có cách nào

đảm bảo rằng một nhân viên quan trọng sẽ không từ chức trong suốt dự án. Thời gian và tiền bạc

sau đó được dành để cố gắng lấp đầy vị trí còn trống và tích hợp người thay thế vào

nhóm, hoặc trong việc tổ chức lại các thành viên còn lại trong nhóm để bù đắp cho sự mất mát. Hoặc

đường đi, lịch trình trượt và ước tính không ổn

Cơ bản của vấn đề ước tính chi phí là một vấn đề khác: Kích thước của một sản phẩm như thế nào để

được đo lường?

9.2.1 Các thước đo về kích thước của một sản phẩm

Số liệu phổ biến nhất cho kích thước của sản phẩm là số dòng mã. Hai đơn vị thường được sử dụng: dòng mã (LOC) và hàng nghìn hướng dẫn nguồn được phân phối (KDSI). Nhiều vấn đề liên quan đến việc sử dụng các dòng mã [van der Poel và Schach, 1983].

* Tạo mã nguồn chỉ là một phần nhỏ trong tổng số nỗ lực phát triển phần mềm.Có vẻ hơi xa vời rằng thời gian cần thiết cho các yêu cầu, phân tích, quy trình công việc thiết kế, triển khai và kiểm tra (bao gồm các hoạt động lập kế hoạch và tài liệu) có thể chỉ được thể hiện dưới dạng một hàm của số dòng mã trong sản phẩm cuối cùng.
* Việc triển khai cùng một sản phẩm bằng hai ngôn ngữ khác nhau dẫn đến các phiên bản có số dòng mã khác nhau. Ngoài ra, với các ngôn ngữ như Lisp hoặc với nhiều 4GL phi thủ tục (Phần 15.2), khái niệm về một dòng mã không được định nghĩa.
* Thường thì không rõ ràng chính xác về cách đếm các dòng mã. Chỉ nên thực thi các dòng mã được đếm hoặc định nghĩa dữ liệu là tốt? Và những bình luận có nên được tính không? Nếu không, có một nguy cơ là các lập trình viên sẽ miễn cưỡng dành thời gian cho những gì họ nhận thấy là những nhận xét "không có lợi", nhưng nếu các nhận xét được tính, thì nguy cơ ngược lại là các lập trình viên sẽ viết hàng loạt nhận xét để cố gắng tăng năng suất. Ngoài ra, những gì về đếm các câu lệnh ngôn ngữ kiểm soát công việc? Một vấn đề khác là cách tính các dòng đã thay đổi hoặc các dòng đã xóa — trong quá trình cải tiến một sản phẩm để cải thiện hiệu suất của nó, đôi khi số lượng dòng mã bị giảm xuống. Việc sử dụng lại mã (Phần 8.1) cũng làm phức tạp việc đếm dòng: Nếu mã được sử dụng lại được sửa đổi, nó được tính như thế nào? Và, điều gì sẽ xảy ra nếu mã được kế thừa từ một lớp cha (Phần 7.8)? Ở trong ngắn gọn, số liệu rõ ràng đơn giản của các dòng mã là bất cứ điều gì nhưng đơn giản để đếm.
* Không phải tất cả các mã được triển khai đều được chuyển đến máy khách. Nó không phải là hiếm đối với một nửa, mã bao gồm các công cụ cần thiết để hỗ trợ nỗ lực phát triển.
* Giả sử rằng một nhà phát triển phần mềm sử dụng trình tạo mã, chẳng hạn như trình tạo báo cáo, trình tạo màn hình hoặc trình tạo giao diện người dùng đồ họa (GUI). Sau một vài phút của hoạt động thiết kế từ phía nhà phát triển, công cụ này có thể tạo ra hàng nghìn dòng mã.
* Số lượng dòng mã trong sản phẩm cuối cùng chỉ có thể được xác định khi sản phẩm hoàn toàn là fi nished. Do đó, ước tính chi phí dựa trên các dòng mã là nguy hiểm gấp đôi. Để bắt đầu quá trình ước tính, số dòng mã trong thành phẩm phải được ước tính. Sau đó, ước tính này được sử dụng để ước tính chi phí sản phẩm. Không chỉ có sự không chắc chắn trong mọi kỹ thuật chi phí, mà nếu đầu vào cho bản thân công cụ ước tính chi phí không chắc chắn cũng không chắc chắn (nghĩa là số dòng mã trong sản phẩm chưa được chế tạo), thì độ tin cậy của ước tính chi phí kết quả là không chắc là rất cao.

Bởi vì số lượng dòng mã không đáng tin cậy, các chỉ số khác phải được xem xét. Một cách tiếp cận thay thế để ước tính kích thước của sản phẩm là sử dụng các chỉ số dựa trên các đại lượng đo lường có thể được xác định sớm trong quá trình phần mềm. Ví dụ, van der Poel và Schach [1983] đã đưa ra số liệu FFP để ước tính chi phí của các sản phẩm xử lý dữ liệu quy mô trung bình. Ba yếu tố cấu trúc cơ bản của xử lý dữ liệu sản phẩm là các tệp, luồng và quy trình của nó; tên FFP là một từ viết tắt được hình thành từ các chữ cái đầu của các phần tử đó. Một fi le được định nghĩa là một tập hợp các hồ sơ liên quan thường trú trong sản phẩm; giao dịch và các tệp tạm thời là bị loại trừ. Một fl ow là một giao diện dữ liệu giữa sản phẩm và môi trường, chẳng hạn như màn hình hoặc một báo cáo. Một quy trình là một thao tác logic hoặc số học được xác định về mặt chức năng dữ liệu; các ví dụ bao gồm sắp xếp, xác thực hoặc cập nhật. Với số lượng tệp Fi, các luồng Fl, và xử lý Pr trong một sản phẩm, kích thước S và chi phí C của nó được cho bởi:

|  |
| --- |
|  |

trong đó d là hằng số thay đổi giữa các tổ chức. Hằng số d là thước đo hiệu quả (năng suất) của quá trình phát triển phần mềm trong đó cơ quan. Kích thước của một sản phẩm đơn giản là tổng số tệp, số lượng tệp và quy trình, một số lượng có thể được xác định sau khi thiết kế kiến trúc hoàn thành. Các khi đó chi phí tỷ lệ với quy mô, hằng số tỷ lệ d được xác định bởi bình phương nhỏ nhất phù hợp với dữ liệu chi phí liên quan đến các sản phẩm do tổ chức đó phát triển trước đó. Không giống như các chỉ số dựa trên số dòng mã, chi phí có thể được ước tính trước

mã hóa bắt đầu. Tính hợp lệ và độ tin cậy của chỉ số FFP đã được chứng minh bằng cách sử dụng mẫu bao gồm một loạt các ứng dụng xử lý dữ liệu quy mô trung bình. Không may, chỉ số này không bao giờ được mở rộng để bao gồm cơ sở dữ liệu, một thành phần thiết yếu của nhiều sản phẩm xử lý cơ sở dữ liệu. Một số liệu tương tự, nhưng được phát triển độc lập, cho kích thước của sản phẩm đã được phát triển của Albrecht [1979] dựa trên các điểm chức năng; Số liệu của Albrecht dựa trên số lượng các mục đầu vào Inp, các mục đầu ra Out, yêu cầu Inq, tệp chính Maf, và giao diện Inf. Trong nó dạng đơn giản nhất số điểm chức năng FP được cho bởi phương trình:

|  |
| --- |
|  |

Vì đây là thước đo kích thước của sản phẩm nên nó có thể được sử dụng để ước tính chi phí và ước tính năng suất. Công thức (9.3) là phép tính đơn giản quá mức của phép tính ba bước. Đầu tiên, không điều chỉnh các điểm chức năng được tính: