

BÀI GIẢNG

QUẢN LÝ DỰ ÁN PHẦN MỀM

CHƯƠNG 2. ƯỚC LƯỢNG DỰ ÁN

NỘI DUNG:

- Giới thiệu
- Các phương pháp ước lượng
- Độ đo

Giới thiệu:

- Ước lượng dự án hiện là khâu yếu nhất hiện nay. Không ước lượng được thì dự án rất dễ vỡ kế hoạch về thời gian và tài chính.
- Thực tế không dự án nào có thể ước lượng chính xác, ước lượng cần được thực hiện nhiều vòng. Mức ước lượng trong giai đoạn xác định có thể sai tới 50-100%, nhưng trong giai đoạn thiết kế phải giảm tới 25-50%, còn trong giai đoạn thiết kế chi tiết chỉ còn 10-25%.
- Ước lượng chỉ có thể chính xác nếu phân rã được các vấn đề nhỏ hơn, đó là kỹ thuật chia để trị (divide and conquer).

Lý do để ước lượng:

- Ước lượng cho chúng ta ý tưởng về thời gian để hoàn thành từng nhiệm vụ và toàn bộ dự án.
- Cho phép xác định trước mức độ tài nguyên cần hoàn thành từng nhiệm vụ và toàn bộ dự án (con người, trang thiết bị).
- Dùng ước lượng để xây dựng lịch biểu, xác định nhiệm vụ nào là quan trọng hơn.
- Dùng ước lượng để đánh giá hiệu năng.

Các phương pháp ước lượng:

Sử dụng đánh giá chuyên gia:

- Các chuyên gia đã có kinh nghiệm triển khai dự án phần mềm, có thể trả lời ngay các ước lượng tuy rằng không phải lúc nào độ chính xác cũng đáng tin cậy
- *Ưu điểm*: nhanh và nếu người được hỏi thực sự là một chuyên gia, thì ước lượng của anh ta có thể chính xác đến ngạc nhiên.
- *Nhược điểm*: độ chính xác phụ thuộc vào thời gian chuyên gia bỏ ra để đánh giá. Ước lượng sẽ không thể tin cậy được, nếu chuyên gia đó lại giao cho một người khác thực hiện. Ngoài ra, việc chỉ dựa vào ý kiến và hiểu biết chủ quan của số ít chuyên gia, cũng là một điều nguy hiểm.

Ước lượng dựa vào quy trình lịch sử (kinh nghiệm quá khứ):

- Viết ra mỗi công việc cần bao lâu để hoàn thành và ai là người chịu trách nhiệm. Sau đó, ta có thể so sánh công việc cần đánh giá với những công việc tương tự đã được thực hiện trong quá khứ và đi tới một ước lượng. Trong lập trình, đó có thể là việc sinh ra biểu mẫu đưa vào, sinh báo cáo, tính toán một công thức phức tạp, ...
- Chú ý, việc tái sử dụng lại và ai là người thực hiện các công việc đó

Ước lượng bằng các mô hình ước lượng thực nghiệm. Phải có các tham số về dự án (các độ đo)

Độ đo:

- Khái niệm độ đo: là các chỉ số đặc trưng cho một khía cạnh nào đó. Trong công nghệ phần mềm có độ đo của phần mềm (software metric/software measure), độ đo của dự án (project metric) và độ đo của quy trình phần mềm (process metric).
- Có độ đo trực tiếp và độ đo gián tiếp. Độ đo trực tiếp là độ đo có thể tính đếm trực tiếp không thông qua các độ đo khác (ví dụ độ đo LOC – lines of code), có độ đo gián tiếp là các độ đo tính qua các độ đo khác (ví dụ tỉ lệ lỗi = số lỗi / số dòng mã nguồn)
- Dự án cũng có độ đo, chi phí cho dự án, năng suất của dự án.
- Quy trình phần mềm cũng có độ đo, chẳng hạn tỉ lệ chi phí trung bình cho mỗi giai đoạn phát triển phần mềm đối với quy trình thác nước.

Độ đo LOC – Metric hướng quy mô phần mềm

- LOC (lines of code) hay KLOC (nghìn dòng lệnh). Độ đo này chỉ có thể chính xác sau khi dự án đã kết thúc. Tuy nhiên bằng kinh nghiệm, hoặc bằng thống kê tương tự có thể ước lượng được khối lượng mã nguồn của một phần mềm trước khi kết thúc dự án.
- LOC sau khi kết thúc dự án sẽ được dùng để ước lượng các dự án tương tự sau này.
- Các độ đo dẫn xuất: số lỗi trên KLOC, chi phí trên KLOC, số tài liệu trên KLOC, năng suất số KLOC /người, tháng.
- LOC phụ thuộc vào môi trường lập trình nên khó so sánh giữa các dự án nếu chúng phát triển trên các môi trường lập trình khác nhau.

Điểm chức năng (Function Point) Metric hướng chức năng

- Điểm chức năng (FP) đo độ phức tạp của phần mềm. Quy mô chỉ phản ánh một khía cạnh nhỏ của độ phức tạp, chính chức năng thể hiện độ phức tạp chính xác hơn.
- FP được tính qua 5 yếu tố chính và 14 yếu tố phụ. Các yếu tố chính là:
 - Số user input (số các thành phần dữ liệu đưa vào), số các input được dùng trong các câu hỏi khác nhau được tính riêng rẽ
 - Số user output (xuất hiện trong các report, các màn hình, các thông báo). Các output trong các câu hỏi khác nhau được kể riêng rẽ
 - Số truy vấn (inquiry) của người sử dụng - số input trong các truy vấn online
 - Số lượng file logic (có thể chỉ là một phần của CSDL, có thể tính như một bảng của CSDL) và các file độc lập
 - Số lượng các giao tiếp ngoài: ngoại vi, các hệ thống thông tin khác mà nó giao tiếp
- Mỗi yếu tố trên được gán một trọng số, tùy theo ảnh hưởng của mỗi yếu tố và tùy theo mức độ phức tạp: thường tính theo 3 mức là đơn giản, trung bình và phức tạp.

Ví dụ:

	Tham số đo	Số đo	ĐG	TB	PT	
1	Số input	25	3	4	6	100
2	Số output	30	4	5	7	150
3	Số inquiry	20	3	4	6	120
4	Số file	10	7	10	15	70
5	Số tương tác ngoài	10	5	7	10	70
	Tổng					510

Điểm chức năng (Function Point) 14 yếu tố điều chỉnh phụ

1. Hệ thống đòi hỏi backup và hồi phục tin cậy
2. Đòi hỏi dữ liệu truyền thống
3. Có các chức năng phân tán
4. Hiệu năng là điều quan trọng
5. Yêu cầu sử dụng môi trường mạng
6. Hệ thống đòi hỏi dữ liệu on-line
7. Khi đòi hỏi dữ liệu online, cần nhiều màn hình dữ liệu hoặc nhiều xử lý
8. Master file được cập nhật online
9. Input, output, file, và tính toán online phức tạp
10. Quá trình xử lý bên trong phức tạp
11. Mã được thiết kế để dùng lại
12. Việc chuyển đổi và cài đặt được tính ngay trong thiết kế
13. Hệ thống được thiết kế để có thể cài đặt nhiều lần cho các tổ chức khác nhau
14. Ứng dụng được thiết kế để dễ thay đổi và làm dễ dàng sử dụng cho người dùng

Mỗi F_i được từ 0 tới 5 điểm tùy theo mức độ

$$FP = \text{Điểm của các yếu tố chính} \times [0.65 + \sum F_i / 100]$$

Ước lượng PERT

- Kỹ thuật này yêu cầu làm 3 ước lượng thời gian cho mỗi nhiệm vụ rồi tổ hợp lại cho ra kết quả cuối cùng:
 - Ước lượng có thể nhất (ML) biểu diễn cho thời gian (dòng lệnh) để hoàn thành một nhiệm vụ trong điều kiện bình thường hay hợp lý.
 - Ước lượng lạc quan nhất (MO) biểu diễn cho thời gian (dòng lệnh) để hoàn thành trong điều kiện tốt nhất.
 - Ước lượng bi quan nhất (MP) biểu diễn thời gian (dòng lệnh) trong điều kiện xấu nhất.

- Công thức ước lượng:

$$MO + 4 ML + MP$$

$$ET = \frac{\quad}{6}$$

Ví dụ 1:

- Có 5 công việc và các ước lượng theo giờ tương ứng:

Nhiệm vụ	MO	ML	MP	ET
Xác định nội dung	8	32	50	31.0
Xác định định dạng	2	8	24	9.7
Xác định yêu cầu in	2	8	12	7.7
Xây dựng bản thảo	16	40	80	42.7
Xác định phân phối	2	10	14	9.3
Xác định hỗ trợ xử lý văn bản	1	3	7	3.3
Tổng cộng	31	101	187	103.7

Sau đó, điều chỉnh lại thời gian trông đợi để tính thời gian bị mất và thời gian ngắt quãng (điển hình là 7 – 10 %)

Nhiệm vụ	ET	% điều chỉnh	ET đã điều chỉnh
Xác định nội dung	31.0	1.10	34.1
Xác định định dạng	9.7	1.10	10.7
Xác định yêu cầu in	7.7	1.10	8.5
Xây dựng bản thảo	42.7	1.10	47.0
Xác định phân phối	9.3	1.10	10.2
Xác định hỗ trợ xử lý văn bản	3.3	1.10	3.6
Tổng cộng	103.7		114.1

Biểu mẫu ước lượng hoạt động:

Số hiệu hoạt động	Mô tả hoạt động	Lạc quan nhất	Có thể nhất	Bi quan nhất	Thời gian trông đợi	Nhân tố điều chỉnh năng suất	Thời gian trông đợi đã sửa lại	Số giờ làm việc/ ngày	Thời hạn
...									
Tổng cộng									

Ưu điểm:

- Người ước lượng dự án phải xem xét các trường hợp có thể xảy ra: bình thường, xấu nhất, tốt nhất. Nghĩa là, xét đến tất cả các yếu tố tác động tới ước lượng.
- Khuyến khích trao đổi giữa các thành viên trong nhóm dự án.
- Giúp cho việc lập lịch chi tiết hơn → dễ kiểm soát dự án hơn.

Nhược điểm:

- Tốn nhiều thời gian cho việc ước lượng: bi quan, lạc quan và có thể cho từng công việc. Từ đó, gây ra các hiệu năng khác: tâm lý chán nản, không làm được các công việc khác, ...
- Đòi hỏi số lượng tính toán lớn, đặc biệt đối với các dự án có số lượng nhiệm vụ nhiều (hàng nghìn)
- Tuy nhiên, các nhược điểm này có thể khắc phục được.

Ví dụ 2:

Giả sử có một bệnh viện cần lắp đặt hệ thống mạng LAN gồm khoảng 70 máy tính, hãy lập kế hoạch cho dự án đó bao gồm những công việc gì, dùng ước lượng PERT để ước lượng mất bao lâu để hoàn thành công việc?

Ước lượng công sức:

- Ước lượng công sức là kỹ thuật thông thường nhất cho việc xác định chi phí cho dự án.
- Số người – ngày, tháng, năm được áp dụng cho từng nhiệm vụ của dự án. Chi phí được gắn với từng đơn vị công sức và suy ra được chi phí ước lượng.
- Ước lượng công sức được phân theo các giai đoạn phát triển phần mềm: xác định yêu cầu, phân tích, thiết kế, thực hiện, kiểm thử và vận hành.

Bảng ước lượng công sức:

	Nhiệm vụ: Xác định yêu cầu, phân tích, ...	Tổng cộng
Chức năng		
Tổng cộng: - Tỷ lệ - Giá thành:		

Ví dụ: Ước lượng công sức (người/tháng):

Chức năng	Phân tích yêu cầu	Thiết kế	Thực hiện	Kiểm thử	Tổng cộng
Kiểm soát giao diện	1.0	2.0	0.5	3.5	7.0
Phân tích hình học 2 chiều	2.0	10.0	4.5	9.5	26.0
Phân tích hình học 3 chiều	2.5	12.0	6.0	11.0	31.5
Quản trị cấu trúc dữ liệu	2.0	6.0	3.0	4.0	15
Hiển thị đồ họa máy tính	1.5	11.0	4.0	10.5	27
Điều khiển ngoại vi	1.5	6.0	3.5	5.0	16
Phân tích thiết kế	4.0	14.0	5.0	7.0	30
Tổng cộng:	14.5	61	26.5	50.5	152.5
- Lương/tháng (\$):	5200	4800	4250	4500	
- Chi phí (\$):	75400	292800	112625	227250	708075

Ước lượng việc lập trình:

- Hai nhân tố ảnh hưởng tới thời gian để thực hiện một công việc: độ phức tạp của công việc (C) và hiệu năng của người thực hiện.
- Hiệu năng của người phụ thuộc vào số năm kinh nghiệm nói chung (G) và tri thức về một công việc đã cho (J).

- Công thức:

$$D = C \times (G + J)$$

- Với:
 - D là độ dài thời gian thực hiện công việc
 - C là nhân tố độ phức tạp
 - G là nhân tố kinh nghiệm nói chung
 - J là nhân tố tri thức về công việc đang xét

Độ phức tạp:

- Độ phức tạp của chương trình phụ thuộc vào ngôn ngữ được dùng và độ phức tạp của từng thời điểm chức năng (hiển thị, điều khiển thiết bị ngoại vi, tính toán, ...)

Ngôn ngữ	Điểm chức năng	Đơn giản	Phức tạp	Rất pt
Bộ thông dịch	Người dùng đưa vào	1	3	4
	Người dùng hiển thị	1	3	4
	Thiết bị ngoại vi vào	3	6	8
	Thiết bị ngoại vi ra	3	6	8
	Cấu trúc lại dữ liệu	1	3	4
	Kiểm tra điều kiện	1	3	4
	Tính toán	1	2	3
	Nhảy	1	2	3
	Gọi	1	2	4

Ngôn ngữ	Điểm chức năng	Đơn giản	Phức tạp	Rất pt
Cấp cao	Người dùng đưa vào	2	4	5
	Người dùng hiển thị	2	4	5
	Thiết bị ngoại vi vào	4	7	9
	Thiết bị ngoại vi ra	4	7	9
	Cấu trúc lại dữ liệu	2	4	5
	Kiểm tra điều kiện	2	4	5
	Tính toán	2	3	4
	Nhảy	1	2	3
	Gọi	1	2	3

Ngôn ngữ	Điểm chức năng	Đơn giản	Phức tạp	Rất pt
Hợp ngữ	Người dùng đưa vào	4	5	8
	Người dùng hiển thị	4	5	8
	Thiết bị ngoại vi vào	6	8	10
	Thiết bị ngoại vi ra	6	8	10
	Cấu trúc lại dữ liệu	4	5	8
	Kiểm tra điều kiện	4	7	9
	Tính toán	3	5	8
	Nhảy	3	4	6
	Gọi	4	5	8

Hiệu năng:

- Thiết lập các nhân tố cho tính hiệu năng của đội ngũ nhân viên của mình. Điều này còn khó hơn nhiều việc tính các nhân tố độ phức tạp công việc, vì hiệu năng của con người có thể thay đổi tùy theo mức độ quan tâm của họ, thái độ,
- Hiệu năng chịu ảnh hưởng bởi những năm kinh nghiệm nói chung và hiểu biết về công việc.

Nhân tố hiệu năng dựa trên năm kinh nghiệm nói chung (G):
(dựa trên công bố của IBM)

Kiểu người lập trình	Năm kinh nghiệm	Phạm vi nhân tố
Cấp cao	5 +	0.5 - 0.75
Trung bình	1.5 - 5	1.0 - 1.5
Tập sự	0.5 - 1.5	2.0 - 3.0
Học nghề	0.0 - 0.5	3.5 - 4.0

Bảng xác định tri thức về công việc:

Tri thức về công việc	Tri thức cần có		
	Nhiều	Vừa	Không cần
Tri thức chi tiết về việc này và tri thức chi tiết về những việc liên quan	0.75	0.25	0.0
Tri thức tốt về việc này và khá về việc liên quan	1.25	0.50	0.0
Tri thức khá về việc này và không biết về các việc liên quan	1.50	0.75	0.0
Không biết về việc này và biết chi tiết về các việc liên quan	2.00	1.25	0.25
Không biết về việc này và không biết về các việc liên quan			

Ví dụ, ước lượng phải mất bao nhiêu lâu để viết một chương trình PASCAL thực hiện chức năng sau:

- (1) Chương trình nhắc người sử dụng về việc gì đó,
- (2) Đọc phản ứng của người sử dụng,
- (3) Kiểm chứng nó
- (4) Đọc một bản ghi từ đĩa
- (5) Tính một số,
- (6) Ghi một bản ghi lên đĩa,
- (7) Hiển thị kết quả cho người sử dụng,
- (8) Gọi một đơn thể khác (quay lại),
- (9) Người lập trình có hai năm kinh nghiệm
- (10) Tri thức khá về ứng dụng đặc biệt nhưng không có tri thức về các ứng dụng có liên quan

Tính độ phức tạp cho ngôn ngữ cấp cao:

Chức năng	Nhân tố
(1) Người dùng hiển thị (đơn giản)	2
(2) Người dùng đưa vào (đơn giản)	2
(3) Kiểm tra điều kiện (phức tạp)	4
(4) Thiết bị ngoại vi vào (đơn giản)	4
(5) Tính toán (đơn giản)	2
(6) Thiết bị ngoại vi ra (đơn giản)	4
(7) Người dùng hiển thị (đơn giản)	2
(8) Gọi (đơn giản)	3
Tổng độ phức tạp (C)	23

- Kinh nghiệm chung G:

- (9) Người lập trình trung bình (2 năm kinh nghiệm) $G = 1.00$

- Tri thức về công việc J:

- (10) Tri thức khá về ứng dụng, không có tri thức liên quan $J = 0.75$

- Vận dụng công thức, ta được:

$$\text{Thời gian} = 23 \times (1.00 + 0.75) = 40.25$$

- Như vậy, nếu sử dụng người này, ta sẽ cần 40 giờ để thiết kế, làm tư liệu, lập trình và kiểm thử chương trình.

* Một số chú ý khi ước lượng:

- Khi ước lượng quá cao:

- Nên kiểm chứng ước lượng của mình bằng việc tìm kiếm ý kiến thứ hai hoặc dựa vào quy trình lịch sử của các dự án khác.
- Thu hẹp phạm vi dự án: Phát triển một sản phẩm là phiên bản nhỏ hơn bản thiết kế gốc (được sự chấp thuận của khách hàng).

- Khi ước lượng quá thấp:

- Các ước lượng thấp thường là kết quả của sự lạc quan của người ước lượng.
- Nên kiểm chứng ước lượng của mình bằng việc tìm kiếm ý kiến thứ hai hoặc dựa vào quy trình lịch sử của các dự án khác.
- Tăng thêm mọi ước lượng với số phần trăm nào đó.
- Yêu cầu người đưa ra ước lượng ký vào một tài liệu chỉ ra cam kết tuân thủ với các ước lượng của họ.

Các bước để thu được ước lượng:

1. Xác định các công việc trong dự án
2. Nhận diện các khoản mục cho từng công việc trong dự án để làm ước lượng cho từng khoản mục đó.
3. Xây dựng biểu mẫu hướng dẫn toàn bộ tiến trình ước lượng.
4. Nhận diện người sẽ làm ước lượng cho từng nhiệm vụ.
5. Tổ chức các phiên họp với những người ước lượng.
6. Thực hiện tính toán
7. Tổ chức phiên họp nhóm với tất cả những người tham gia ước lượng.
8. Làm những sửa đổi cho những ước lượng và ghi những ước lượng này, lưu tài liệu.
9. Mọi người ký vào tài liệu
10. Phân phát tài liệu biên soạn mà mọi người đã ký, đảm bảo tính minh bạch và mọi người làm việc theo các ước lượng đó.

Mô hình ước lượng thực nghiệm:

- Hầu hết các mô hình thực nghiệm đều phải có đầu vào từ một độ đo độ phức tạp của dự án có thể là LOC hay FP.
- Mô hình chính $E = A + B \times V^C$ trong đó E là công sức có thể đo bằng người tháng, B và C là một hằng số đặc trưng cho mô hình và V là LOC hay FP. Mô hình này cho thấy công sức không tỉ lệ tuyến tính theo độ phức tạp.
- Một vài ước lượng:
 - Waston Felix $E = 5.2 + KLOC^{0.91}$
 - Baley Basil $E = 5.5 + 0.73 KLOC^{1.16}$
 - Matson Barret $E = 585.7 + 15.12 \times FP$

Kết luận:

- Ước lượng, như tiêu đề đã gợi ý, là khả năng yếu nhất của chúng ta. Thế mà tất cả mọi kế hoạch và kiểm soát của ta đều phụ thuộc vào kỹ năng ước lượng.
- Ước lượng là một quá trình lặp - nó cần phải được hiệu chỉnh dần. Đó là lý do vì sao ta phải có các sự kiện mốc trong quá trình xây dựng một sản phẩm. Sự kiện mốc cho phép ta dừng lại, tính toán xem cần bao lâu để đạt tới đó và ước lượng lại ngày tháng cho các sự kiện mốc tiếp theo dựa trên kinh nghiệm đã có.
- Ước lượng vẫn còn là một nghệ thuật. Không tồn tại phần mềm hay công cụ nào có sẵn để giúp chúng ta làm việc này. Điều mấu chốt cho việc ước lượng là chia nhỏ. Kinh nghiệm, thống kê cũng rất có ích. Nếu công việc được chia thành những phần nhỏ hơn và bạn tính toán ước lượng cho từng phần việc nhỏ, một số phần sẽ có thể được ước lượng thừa, một số phần khác bị ước lượng thiếu. Cuối cùng bạn phải lấy trung bình - và đó cũng là điểm duy nhất có vấn đề.