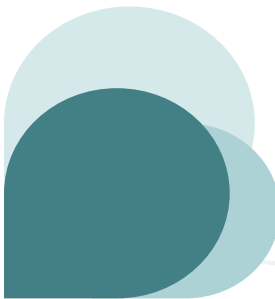


PHẦN III: PHƯƠNG PHÁP XÁC ĐỊNH YÊU CẦU NGƯỜI DÙNG

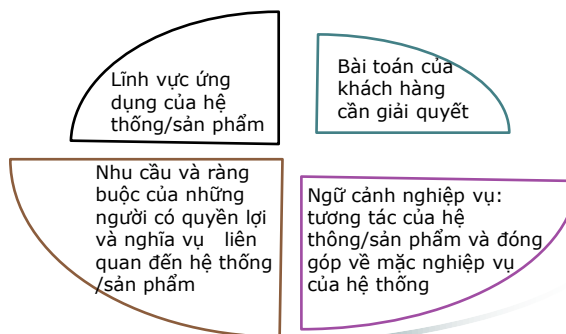
- 
- I. Tổng quan về yêu cầu phần mềm
 - II. Quy trình xác định yêu cầu phần mềm
 - III. Công cụ và phương pháp đặc tả yêu cầu phần mềm
 - IV. Nguyên lý phân tích yêu cầu sử dụng

1

1. Khái niệm

I. Tổng quan về yêu cầu phần mềm

- Các đặc tính của hệ thống hay sản phẩm do khách hàng - người sử dụng phần mềm - nêu ra
→ Xác định được phần mềm đáp ứng được các yêu cầu và mong muốn của khách hàng - người sử dụng phần mềm



2



I. Tổng quan về yêu cầu phần mềm

Tại sao cần phải đặt ra yêu cầu phần mềm ?

- Khách hàng chỉ có những ý tưởng còn mơ hồ về phần mềm cần phải xây dựng để phục vụ công việc của họ, chúng ta phải sẵn sàng, kiên trì theo đuổi để đi từ các ý tưởng mơ hồ đó đến "Phần mềm có đầy đủ các tính năng cần thiết"
- Khách hàng rất hay thay đổi các đòi hỏi của mình, chúng ta nắm bắt được các thay đổi đó và sửa đổi các mô tả một cách hợp lý

3



I. Tổng quan về yêu cầu phần mềm

2. Phân loại

- Theo 4 thành phần của phần mềm:
- Theo cách đặc tả phần mềm

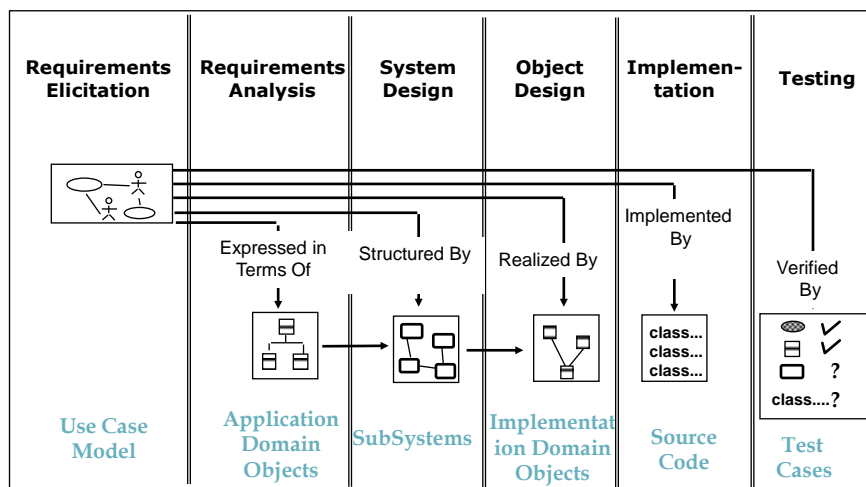
4

II. Quy trình xác định yêu cầu PM

- Phát hiện các yêu cầu phần mềm (Requirements elicitation)
- Phân tích các yêu cầu phần mềm và thương lượng với khách hàng (Requirements analysis and negotiation)
- Đặc tả các yêu cầu phần mềm (Requirements specification)
- Mô hình hóa hệ thống (System modeling)
- Kiểm tra tính hợp lý của các yêu cầu phần mềm (Requirements validation)
- Quản trị các yêu cầu phần mềm (Requirements management)

5

Ví dụ: Quy trình xác định yêu cầu phần mềm hướng đối tượng



Or textual requirements

6

1. Phát hiện yêu cầu phần mềm

- Đánh giá tính khả thi về kỹ thuật và nghiệp vụ của phần mềm định phát triển
- Tìm kiếm các nhân sự (chuyên gia, người sử dụng) có những hiểu biết sâu sắc nhất, chi tiết nhất về hệ thống giúp chúng ta xác định yêu cầu phần mềm
- Xác định môi trường kỹ thuật trong đó sẽ triển khai phần mềm
- Xác định các ràng buộc về lĩnh vực ứng dụng của phần mềm (giới hạn về chức năng/hiệu năng phần mềm)

7

1. Phát hiện yêu cầu phần mềm

- Xác định các phương pháp sử dụng để phát hiện các yêu cầu phần mềm: phỏng vấn, làm việc nhóm, các buổi họp, gặp gỡ đối tác, v.v.
- Thu hút sự tham gia của nhiều chuyên gia, khách hàng để chúng ta có được các quan điểm xem xét phần mềm khác nhau từ phía khách hàng
- Xác định các yêu cầu còn nhập nhằng để làm mẫu thử
- Thiết kế các kịch bản sử dụng của phần mềm để giúp khách hàng định rõ các yêu cầu chính.

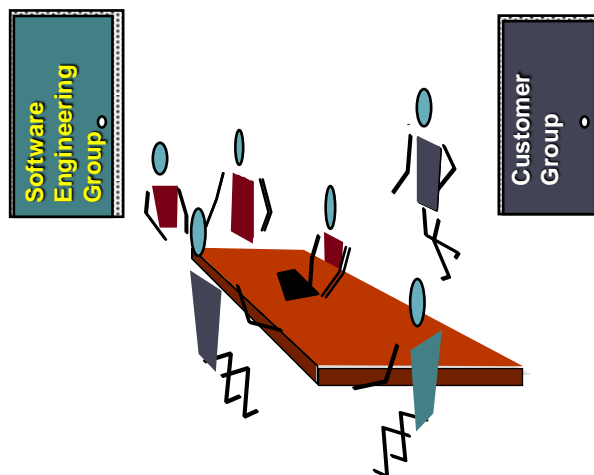
8

Đầu ra của bước phát hiện yêu cầu phần mềm

- Bảng kê (statement) các đòi hỏi và chức năng khả thi của phần mềm
- Bảng kê phạm vi ứng dụng của phần mềm
- Mô tả môi trường kỹ thuật của phần mềm
- Bảng kê tập hợp các kịch bản sử dụng của phần mềm
- Các nguyên mẫu xây dựng, phát triển hay sử dụng trong phần mềm (nếu có)
- Danh sách nhân sự tham gia vào quá trình phát hiện các yêu cầu phần mềm - kể cả các nhân sự từ phía công ty- khách hàng

9

2. Phân tích các yêu cầu PM và thương lượng với khách hàng



10



2. Phân tích các yêu cầu PM và thương lượng với khách hàng

- Phân loại các yêu cầu phần mềm và sắp xếp chúng theo các nhóm liên quan
- Khảo sát tỉ mỉ từng yêu cầu phần mềm trong mối quan hệ của nó với các yêu cầu phần mềm khác
- Thẩm định từng yêu cầu phần mềm theo các tính chất: phù hợp, đầy đủ, rõ ràng, không trùng lặp
- Phân cấp các yêu cầu phần mềm theo dựa trên nhu cầu và đòi hỏi khách hàng / người sử dụng
- Thẩm định từng yêu cầu phần mềm để xác định chúng có khả năng thực hiện được trong môi trường kỹ thuật hay không, có khả năng kiểm định các yêu cầu phần mềm hay không

11



2. Phân tích các yêu cầu PM và thương lượng với khách hàng

- Thẩm định các rủi ro có thể xảy ra với từng yêu cầu phần mềm
- Đánh giá thô (tương đối) về giá thành và thời gian thực hiện của từng yêu cầu phần mềm trong giá thành sản phẩm phần mềm và thời gian thực hiện phần mềm
- Giải quyết tất cả các bất đồng về yêu cầu phần mềm với khách hàng / người sử dụng trên cơ sở thảo luận và thương lượng các yêu cầu đề ra

12



3. Đặc tả yêu cầu phần mềm

- Đặc tả các yêu cầu phần mềm: xây dựng các tài liệu đặc tả, trong đó có thể sử dụng tới các công cụ như: mô hình hóa, mô hình toán học hình thức (a formal mathematical model), tập hợp các kịch bản sử dụng, các nguyên mẫu hoặc bất kỳ một tổ hợp các công cụ nói trên
- Phương pháp đặc tả:
 - Đặc tả phi hình thức (Informal specifications):
 - Đặc tả hình thức (Formal specifications):
- Tiêu chí đánh giá chất lượng của hồ sơ đặc tả:
 - Tính rõ ràng, chính xác
 - Tính phù hợp
 - Tính đầy đủ, hoàn thiện

13



Ví dụ: Các yêu cầu về hồ sơ đặc tả

- Đặc tả hành vi bên ngoài của HT
- Đặc tả các ràng buộc về cài đặt
- Dễ thay đổi
- Dùng như công cụ tham khảo cho bảo trì
- Sự ghi chép cẩn thận về vòng đời của HT, nghĩa là dự đoán các thay đổi
- Các đáp ứng với các sự cố không mong đợi

14



3.1. Các thành phần của hồ sơ đặc tả

- Đặc tả vận hành hay đặc tả chức năng (Operational specifications) mô tả các hoạt động của hệ thống phần mềm sẽ xây dựng:
- Đặc tả mô tả hay đặc tả phi chức năng (Descriptive specifications): đặc tả các đặc tính, đặc trưng của phần mềm:
- Ngoài ra còn có yêu cầu về lĩnh vực, bắt nguồn từ lĩnh vực của ứng dụng hệ thống và các đặc trưng của lĩnh vực này.

15



Đặc tả chức năng

- Miêu tả các chức năng của hệ thống, phụ thuộc vào kiểu phần mềm và mong đợi của người dùng
 - Tương tác giữa phần mềm và môi trường, độc lập với việc cài đặt
- Các công cụ đặc tả tiêu biểu:
 - Biểu đồ luồng dữ liệu (Data Flow Diagrams)
 - Máy trạng thái hữu hạn (Finite State Machines)
 - Mạng Petri (Petri nets),...
 - Tuy nhiên không bắt buộc và có thể dùng ngôn ngữ tự nhiên.

16



Đặc tả phi chức năng và ràng buộc

- Yêu cầu phi chức năng: Định nghĩa các khía cạnh sử dụng phần mềm, không liên quan trực tiếp tới các hành vi chức năng:
- Ràng buộc: do khách hàng hay môi trường thực thi phần mềm đặt ra
- Thường sử dụng các công cụ
 - Biểu đồ thực thể liên kết (Entity-Relationship Diagrams)
 - Đặc tả Logic (Logic Specifications)
 - Đặc tả đại số (Algebraic Specifications)

17

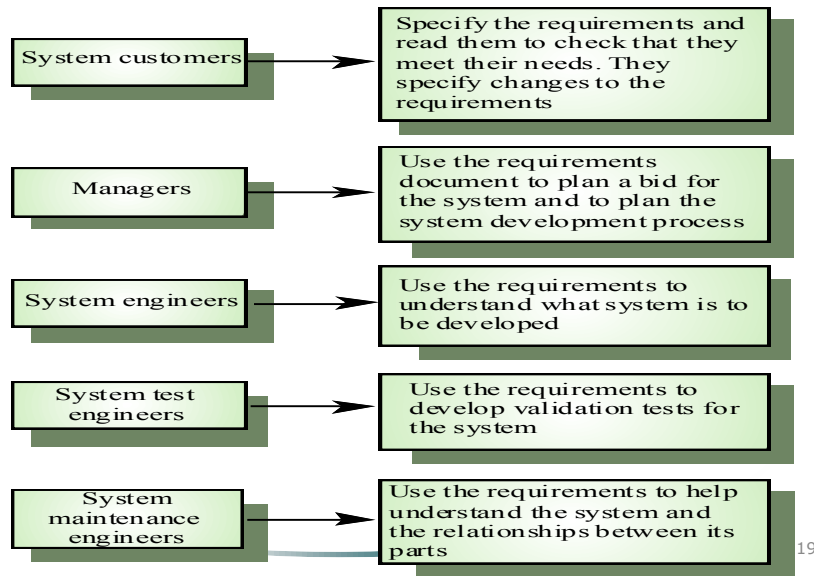


3.2. Tài liệu yêu cầu

- Tài liệu về yêu cầu là các phát biểu chính thức về cái được yêu cầu bởi các nhà phát triển HT
- Nó bao gồm cả 2 phần: định nghĩa và đặc tả yêu cầu
- Nó không phải là tài liệu thiết kế. Tốt hơn có thể nó chỉ là 1 tập các cái mà HT phải làm hơn là HT phải làm thế nào (PT chứ không phải là TK)

18

Nội dung cần có của tài liệu yêu cầu



19

III. Phương pháp và công cụ đặc tả yêu cầu phần mềm

- I. Tổng quan về yêu cầu phần mềm
- II. Quy trình xác định yêu cầu phần mềm

- Biểu đồ phân cấp chức năng - WBS (work break down structure)
- Biểu đồ luồng dữ liệu - DFD (data flow diagram)
- Máy trạng thái - FSM (Finite state machine)
- Sơ đồ thực thể liên kết - ERD (entity relation diagram)

20



1. Đặc tả chức năng với DFD

- Hệ thống (System): tập hợp các dữ liệu (data) được xử lý bằng các chức năng tương ứng (functions)
- Các ký pháp sử dụng:



Thể hiện các chức năng (functions)



Thể hiện luồng dữ liệu



Kho dữ liệu

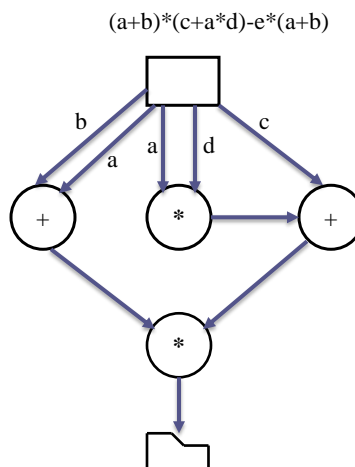


Vào ra dữ liệu và tương tác giữa hệ thống và người sử dụng

21

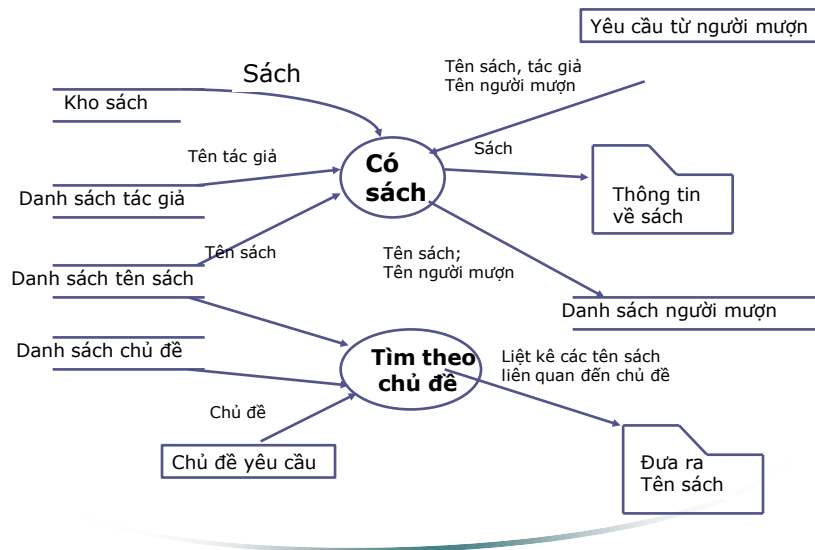


Ví dụ: mô tả biểu thức toán học bằng DFD



22

Ví dụ đặc tả các chức năng của thư viện qua DFD



23

Các hạn chế của DFD

- Ý nghĩa của các ký pháp sử dụng được xác định bởi các định danh lựa chọn của NSD
- Ví dụ: DFD của chức năng tìm kiếm sách:

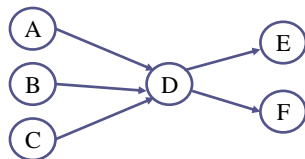

```

      If NSD nhập vào cả tên tác giả và tiêu đề sách Then
        tìm kiếm sách tương ứng, không có thì thông báo lỗi
      Elseif chỉ nhập tên tác giả Then
        hiển thị danh sách các sách tương ứng với
        tên tác giả đã nhập và yêu cầu NSD lựa chọn sách
      Elseif chỉ nhập tiêu đề sách Then
        ...
      Endif
      
```

24

Các hạn chế của DFD

- Trong DFD không xác định rõ các hướng thực hiện (control aspects)



- Biểu đồ DFD này không chỉ rõ đầu vào là gì để thực hiện chức năng D và đầu ra là gì sau khi thực hiện chức năng D.

- Chức năng D có thể cần cả A, B và C
- Chức năng D có thể chỉ cần một trong A, B và C để thực hiện
- Chức năng D có thể kết xuất kết quả cho một trong E và F
- Chức năng D có thể kết xuất kết quả chung cho cả E và F
- Chức năng D có thể kết xuất kết quả riêng cho cả E và F

25

Các hạn chế của DFD

- DFD không xác định sự đồng bộ giữa các chức năng / mô-đun
 - A xử lý dữ liệu và B được hưởng (nhận) các kết quả được xử lý từ A
 - A và B là các chức năng không đồng bộ (asynchronous activities) vì thế cần có buffer để ngăn chặn tình trạng mất dữ liệu

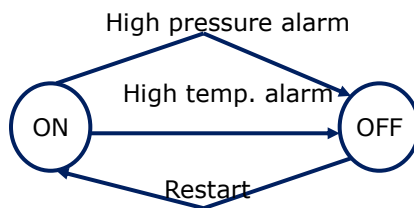


26

2. Đặc tả trạng thái với FSM - Finite State Machines

- FSM chứa
 - Tập hữu hạn các trạng thái Q
 - Tập hữu hạn các đầu vào I
 - Các chức năng chuyển tiếp

$$\delta : Q \times I \rightarrow Q$$



27

Ví dụ: thư viện

- Xét các giao dịch:
 - Mượn sách / Trả sách
 - Thêm đầu sách / Loại bỏ đầu sách
 - Liệt kê danh sách các đầu sách theo tên tác giả hay theo chủ đề
 - Tìm kiếm sách theo các yêu cầu của người mượn
 - Tìm kiếm sách quá hạn trả, . . .

28

Đặc tả các yêu cầu đặc biệt của thư viện

- Độc giả không được mượn quá một số lượng sách nhất định, trong một thời gian nhất định
- Một số sách không được mượn về
- Một số người không được mượn một số loại sách nào đó, . . .

29

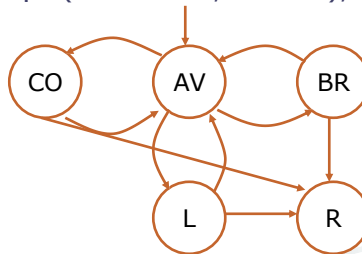
Đặc tả các đối tượng trong thư viện

- Các đối tượng:
 - Tên sách
 - Mã quyển
 - Nhân viên phục vụ
 - Người mượn
- Cần có:
 - tập hợp (danh sách) các tiêu đề sách
 - danh sách các tác giả cho từng quyển sách,
 - danh sách các chủ đề liên quan của các quyển sách

30

FSM đặc tả trạng thái

- Ta có tập hợp các sách (mỗi đầu sách có thể có nhiều quyền sách trong thư viện).
- Mỗi quyền sách có thể có 1 trong 5 trạng thái sau:
 - (AV) Available: được phép mượn,
 - (CO) - (BR): đã mượn (Check Out; Borrow),
 - (L): Last,
 - (R): Remove



Có thể có hạn chế về số sách được mượn cho 1 nhóm độc giả hoặc mọi độc giả, . . .³¹

3. Đặc tả dữ liệu với Mô hình thực thể liên kết -ERD

- Mô hình khái niệm cho phép đặc tả các yêu cầu logic của hệ thống, thường được sử dụng trong các hệ thống dữ liệu lớn
- ER Model
 - Thực thể
 - Quan hệ
 - Thuộc tính
- Biểu đồ thực thể



Thực thể

- Thực thể : tập hợp các thông tin liên quan cần được xử lý trong phần mềm
- Thực thể có thể có mối quan hệ:
 - person owns car



- Thực thể có các thuộc tính

33



Thuộc tính

- Tính chất của một thực thể hoặc một đối tượng dữ liệu
 - đặt tên cho 1 mẫu (instance) của đối tượng dữ liệu
 - mô tả mẫu (instance)
 - tạo liên kết (reference) đến các mẫu khác

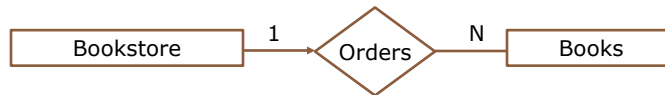


Tập các thuộc tính của 1 đối tượng dữ liệu được xác định thông qua ngữ cảnh của bài toán.

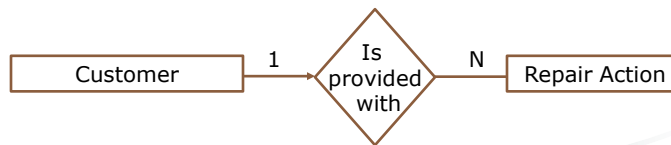
34

Quan hệ

- Chỉ ra mối liên quan giữa các đối tượng dữ liệu

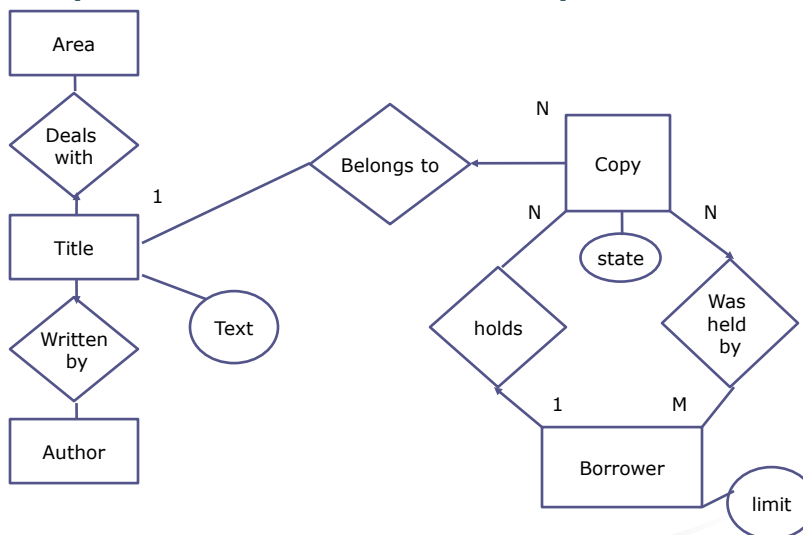


- Cardinality : chỉ ra định lượng của mỗi quan hệ
1:1 one-to-one 1:N one-to-many M:N many-to-many
- Modality : 0 – có thể có, có thể không có quan hệ
1 – bắt buộc có quan hệ



35

Ví dụ: ERD mô tả thư viện



36



4. Thế nào là một đặc tả tốt?

- Dễ hiểu với người dùng
- Có ít điều nhập nhằng
- Có ít quy ước khi mô tả, có thể tạo đơn giản
- Với phong cách từ trên xuống (topdown)
- Dễ triển khai cho những pha sau của vòng đời: thiết kế hệ thống và thiết kế chương trình và giao diện để làm, đảm bảo tính nhất quán, . . .

37



1. Mô hình hóa dữ liệu

- I. Tổng quan về yêu cầu phần mềm
- II. Quy trình xác định yêu cầu phần mềm
- III. Công cụ và phương pháp đặc tả yêu cầu phần mềm
- IV. Nguyên lý phân tích yêu cầu sử dụng**

- Xác định các đối tượng dữ liệu
- Xác định các đặc tính của các đối tượng dữ liệu
- Thiết lập các mối quan hệ giữa các đối tượng dữ liệu

38

2. Mô hình hóa các chức năng

I. Phương pháp xác định yêu cầu

1. Tổng quan về yêu cầu phần mềm
2. Quy trình xác định yêu cầu phần mềm
3. Công cụ và phương pháp đặc tả yêu cầu phần mềm
4. Nguyên lý phân tích yêu cầu sử dụng

- Xác định các chức năng chuyển đổi đối tượng dữ liệu
- Chỉ ra luồng dữ liệu đi qua hệ thống như thế nào
- Biểu diễn bộ phận sản sinh dữ liệu và bộ phận tiêu thụ dữ liệu

39

3. Mô hình hóa hành vi

I. Phương pháp xác định yêu cầu

1. Tổng quan về yêu cầu phần mềm
2. Quy trình xác định yêu cầu phần mềm
3. Công cụ và phương pháp đặc tả yêu cầu phần mềm
4. Nguyên lý phân tích yêu cầu sử dụng

- Chỉ ra các trạng thái (states) khác nhau của hệ thống
- Đặc tả các hiện tượng (events) làm hệ thống thay đổi trạng thái

40

4. Phân mảnh các mô hình

I. Phương pháp xác định yêu cầu

1. Tổng quan về yêu cầu phần mềm
2. Quy trình xác định yêu cầu phần mềm
3. Công cụ và phương pháp đặc tả yêu cầu phần mềm
4. Nguyên lý phân tích yêu cầu sử dụng

- Tinh lọc từng mô hình để biểu diễn các mức trừu tượng thấp hơn
- Lọc đối tượng dữ liệu
- Tạo ra phân cấp chức năng
- Biểu diễn hành vi (behavior) ở các mức chi tiết khác nhau

41

5. Bản chất

I. Phương pháp xác định yêu cầu

1. Tổng quan về yêu cầu phần mềm
2. Quy trình xác định yêu cầu phần mềm
3. Công cụ và phương pháp đặc tả yêu cầu phần mềm
4. Nguyên lý phân tích yêu cầu sử dụng

- Hãy bắt đầu bằng cách tập trung vào bản chất của vấn đề chứ không xem xét những chi tiết cài đặt

42