**MỤC LỤC**

[I. Yêu cầu chung 2](#_Toc445197002)

[1. Mục tiêu 2](#_Toc445197003)

[2. Chuẩn bị 2](#_Toc445197005)

[3. Thành lập nhóm: 2](#_Toc445197006)

[4. Quy định trình bày 2](#_Toc445197007)

[II. Phần chi tiết theo từng buổi bài tập/ thảo luận 3](#_Toc445197014)

[2.1 Buổi 1: Bài tập kiểm thử hộp đen 3](#_Toc445197015)

[2.2 Buổi 2: Bài tập kiểm thử hộp trắng 6](#_Toc445197016)

[2.3 Buổi 3: Bài tập kết hợp giữa kiểm thử hộp đen và kiểm thử hộp trắng 13](#_Toc445197017)

[2.4 Buổi 4: Thảo luận về công cụ kiểm thử QTpro 16](#_Toc445197018)

[2.5 Buổi 5: Thảo luận về công cụ kiểm thử Jmetter 17](#_Toc445197019)

[2.6 Buổi 6: Thảo luận về công cụ kiểm thử JUNIT 17](#_Toc445197020)

[III. Phụ lục 1](#_Toc445197022)9

[PHỤ LỤC 1 – CMM, CMMI 19](#_Toc445197023)

**BÀI TẬP THẢO LUẬN**

**TÊN HỌC PHẦN: KIỂM CHỨNG PHẦN MỀM**

# I. Yêu cầu chung

## 1. Mục tiêu

## Sau khi kết thúc học phần mỗi nhóm sinh viên phải tiến hành kiểm thử thành công một phần mềm dựa trên một công cụ kiểm thử, các bước thực hiện theo đúng qui trình được học. Thành thạo các dạng bài tập cơ bản. Và hoàn thiện toàn bộ các tài liệu, giấy tờ theo đúng biểu mẫu, các bước tương ứng với từng giai đoạn trong quy trình.

## 2. Chuẩn bị

- Giảng viên: Phương tiện dạy học (máy tính, máy chiếu, phấn viết bảng), đề cương môn học, giáo trình, giáo án, tài liệu tham khảo, danh sách các đề tài, bài tập cho sinh viên thảo luận.

- Sinh viên: Đề cương môn học, bài giảng, tài liệu tham khảo, các đề tài thảo luận, bài tập, phương tiện, dụng cụ học tập.

## 3. Thành lập nhóm: (mỗi lớp học chia thành 5 – 6 nhóm sinh viên)

- Sinh viên chuẩn bị nhóm: 08 - 10 sinh viên/1 nhóm.

- Các nhóm có thể đề xuất đề tài thực hiện hoặc theo sự phân công của giảng viên.

- Lập bảng phân công công việc chi tiết của từng thành viên trong nhóm.

- Tại mỗi buổi thảo luận: các nhóm cần trình bày bản báo cáo kết quả đã thực hiện được theo kế hoạch, bao gồm: các mẫu biểu (giấy tờ thực đi kèm), slide báo cáo, demo chương trình.

- Hoàn thiện báo cáo, phần mềm mô phỏng, bộ dữ liệu thử, các kết quả test vào cuối học kỳ: in quyển báo cáo, đĩa CD (slide, báo cáo, demo chương trình).

## 4. Quy định trình bày

## Hình thức trình bày:

* + Font chữ Time new roman, cỡ chữ 13, Spacing: Befor 0pt, After: 6pt, Line spacing: 1.3.
  + Khổ giấy a4, lề trên, dưới, phải 2cm, lề trái 3 cm.

## Bố cục nội dung:

* + Mục lục
  + Lời giới thiệu
  + Bảng phân công công việc của từng thành viên trong nhóm (họ tên, công việc)
  + Bảng danh mục các từ viết tắt (nếu có)
  + Bảng danh mục các hình vẽ (nếu có)
  + Bảng danh mục các biểu đồ (nếu có)

# Phần mở đầu:

* + Tầm quan trọng của vấn đề giải quyết
  + Tên chủ đề lựa chọn, phạm vi liên quan đến vấn đề được đặt ra
  + Quá trình phát triển, hiện trạng vấn đề lựa chọn
  + Vai trò và ý nghĩa của đề tài trong lĩnh vực công nghệ phần mềm , kiểm chứng phần mềm nói riêng và công nghệ thông tin nói chung.

# Nội dung

Tùy thuộc mỗi loại vấn đề mà việc trình bài nội dung có cách bố trí và bố cục

riêng.

* + Đặt vấn đề- đưa ra bài toán cụ thể
  + Phương pháp, công cụ sử dụng
  + Các bước giải quyết
  + Ưu, nhược điểm của cách giải quyết
  + Ví dụ minh họa
  + So sánh với các giải pháp khác cho vấn đề tương tự
  + Các mẫu biểu liên quan

# Kết luận.

* + Sự triển khai ứng dụng hiện tại và triển vọng của giải pháp
  + Những vấn đề còn tồn tại cần nghiên cứu tiếp
  + Những đề xuất có thể

# Tài liệu tham khảo.

# 

# II. Phần chi tiết theo từng buổi bài tập/ thảo luận

## 2.1 Buổi 1: Bài tập kiểm thử hộp đen

**a) Mục tiêu**

Sinh viên nắm vững những kiến thức tổng quan về kiểm chứng phần mềm. Biết vận dụng những kiến thức đã học vào cuộc sống nói chung và thực tế nghề nghiệp nói riêng. Đứng trước một yêu cầu trong thực tế sinh viên có kỹ năng xác định được bài toán, và đưa ra được hướng giải quyết vấn đề.

Nắm vững các các kỹ thuật kiểm thử hộp đen (cụ thể là các kỹ thuật tạo testcase trong kiểm thử hộp đen và các bước tiến hành kiểm thử hộp đen).

**b) Nội dung sinh viên phải biết**

- Sinh viên nắm vững và tổng hợp được những kiến thức cơ bản trong lĩnh vực kiểm chứng phần mềm: Các khái niệm; qui trình quản lý.

- Nắm chắc các bước kiểm thử hộp đen

- Các bước tiến hành tạo testcase ứng với từng kỹ thuật: kỹ thuật chia miền tương đương, kỹ thuật test biên,…

- Áp dụng được vào từng bài tập cụ thể

- Có hiểu biết về xu hướng hiện nay của lĩnh vực kiểm chứng và đảm bảo chất lượng phần mềm.

**c) Nội dung bài tập**

**Bài 1:**

1. Trình bày kiểm thử chia miền tương đương?
2. Áp dụng: Cho một chương trình xác nhận một trường số đếm như sau: Các giá trị nhỏ hơn 10 bị loại bỏ, các giá trị giữa 10 và 21 được chấp nhận, các giá trị lớn hơn hoặc bằng 22 bị loại bỏ.

*Yêu cầu:* Sử dụng phương pháp chia miền tương đương xác định tập testcase bao phủ miền giá trị trên ?

**Bài 2:**

1. Trình bày phương pháp test giá trị biên?
2. Áp dụng: Cho một chương trình xác nhận một trường số đếm như sau: Các giá trị nhỏ hơn 10 bị loại bỏ, các giá trị giữa 10 và 21 được chấp nhận, các giá trị lớn hơn hoặc bằng 22 bị loại bỏ.

*Yêu cầu :* Sử dụng phương pháp test giá trị biên xác định các giá trị đầu vào của các giá trị biên?

**Bài 3:**

1. Trình bày tóm tắt phương pháp test giá trị biên?
2. Trình bày tóm tắt phương pháp chia miền tương đương?
3. Áp dụng: Số thứ tự trên hệ thống điều khiển cổ phiếu có thể sắp xếp từ 10000 đến 99999 (bao gồm cả 10000 và 99999).

*Yêu cầu:* Hãy chỉ ra các đầu vào nào có thể là một kết quả của thiết kế kiểm thử chỉ cho các lớp tương đương đúng và các lớp biên đúng?

**Bài 4:**

1. Trình bày kỹ thuật kiểm thử giá trị biên?
2. Áp dụng: Một hệ thống được thiết kế để đóng thuế như sau: “Một nhân viên có lương £4000 thì được miễn thuế. £1500 tiếp theo bị đánh thuế 10%. £28000 tiếp theo bị đánh thuế 22%. Bất kỳ lượng tiền lớn hơn sau đó đều bị đánh thuế 44%”.

*Yêu cầu:* Giá trị nào là một trường hợp phân tích giá trị biên đúng cho giá trị tiền lương thấp nhất bị đánh phần trăm thuế lớn nhất?

**Bài 5**:

1. Trình bày phương pháp phân hoạch tương đương?
2. Áp dụng: Cho một hệ thống được thiết kế để nộp thuế như sau: “Một nhân viên có lương £4000 thì được miễn thuế. £1500 tiếp theo bị đánh thuế 10%, £28000 tiếp theo bị đánh thuế 22%, lượng tiền cao hơn nữa bị đánh thuế 40%”.

*Yêu cầu:* Xác định testcase bao phủ các lớp tương đương?

**Bài 6:**

1. Trình bày phương pháp phân hoạch tương đương?
2. Áp dụng: Cho chương trình xác định trường số sau:

* Giá trị nhỏ hơn 10 thì loại
* Giá trị từ 10 đến 21 được chấp nhận
* Giá trị lớn hơn hoặc bằng 22 bị loại

*Yêu cầu:* Hãy xác định giá trị đầu vào chứa tất cả các phân hoạch tương đương?

**Bài 7:**

1. Trình bày phương pháp test giá trị biên?
2. Áp dụng: Cho chương trình xác định trường số sau:

* Giá trị nhỏ hơn 10 thì loại
* Giá trị từ 10 đến 21 được chấp nhận
* Giá trị lớn hơn hoặc bằng 22 bị loại

*Yêu cầu:* Xác định giá trị đầu vào chứa các giá trị biên nhiều nhất?

**d) Hướng dẫn/ Gợi ý**

Xem bài tập mẫu trong slide bài giảng. Nội dung các kỹ thuật kiểm thử hộp đen trong chương 4 bài giảng.

**--------------------------------------------------------------------------------------------------------**

## 2.2 Buổi 2: Bài tập kiểm thử hộp trắng

**a) Mục tiêu**

Sinh viên nắm vững những kiến thức tổng quan về kiểm chứng phần mềm. Biết vận dụng những kiến thức đã học vào cuộc sống nói chung và thực tế nghề nghiệp nói riêng. Đứng trước một yêu cầu trong thực tế sinh viên có kỹ năng xác định được bài toán, và đưa ra được hướng giải quyết vấn đề.

Sinh viên nắm được các bước tiến hành kiểm thử hộp trắng, các kỹ thuật kiểm thử hộp trăng (kỹ thuật tạo testcase trong kiểm thử hộp trắng).

**b) Nội dung sinh viên phải biết**

- Sinh viên nắm vững và tổng hợp được những kiến thức cơ bản trong lĩnh vực kiểm chứng phần mềm: Các khái niệm; qui trình quản lý.

- Nắm chắc các bước kiểm thử hộp trắng.

- Các bước tiến hành tạo testcase ứng với từng kỹ thuật: kỹ thuật kiểm thử đồ thị luồng điều khiển, kỹ thuật kiểm thử đồ thị luồng dữ liệu,…

- Áp dụng được vào từng bài tập cụ thể

- Có hiểu biết về xu hướng hiện nay của lĩnh vực kiểm chứng và đảm bảo chất lượng phần mềm.

**c) Nội dung bài tập**

**Bài 1:** Cho đoạn chương trình:

*int i,j,kt;*

*for(i=1;i<=100;i++)*

*{*

*for(j=2;j<i/2;j++)*

*{*

*kt=1;*

*if(i%j==0)*

*{*

*kt=0;*

*break;*

*}*

*}*

*if(kt==1) printf("%4d",i);*

1. Vẽ sơ đồ luồng điều khiển cho đoạn chương trình trên?
2. Tính hằng số Cyclomatic (C)?
3. Liệt kê 1 tập gồm C đường cơ bản?
4. Sinh tập testcase tương ứng?

**Bài 2:** Cho đoạn chương trình sau:

*int i;*

*printf("nhap so phan tu cua mang ");*

*scanf("%d",n);*

*\*a=(int \*)malloc((\*n)\*sizeof(int));*

*for (i=0; i< \*n; i++)*

*{*

*printf("a[%d]= ",i+1);*

*scanf("%d",\*a+i);*

*}*

1. Vẽ sơ đồ luồng điều khiển cho đoạn chương trình trên?
2. Có tối thiểu bao nhiêu bao phủ câu lệnh? Liệt kê các đường đó?
3. Có tối thiểu bao nhiêu bảo phủ nhánh? Liệt kê các đường đó?

**Bài 3:** Cho đoạn chương trình sau:

*for (int i=0; i<n; i++)*

*printf("%d ",\*(a+i));*

1. Vẽ sơ đồ luồng điều khiển cho đoạn chương trình trên?
2. Có tối thiểu bao nhiêu bao phủ câu lệnh? Liệt kê các đường đó?
3. Có tối thiểu bao nhiêu bảo phủ nhánh? Liệt kê các đường đó?

**Bài 4:** Cho đoạn chương trình sau:

*void vechuot(int x,int y, int color, int press, int &count1, int &count2)*

*{*

*textcolor(color);*

*gotoxy(x/8+1,y/8+1);*

*cprintf("^");*

*if (press==1) cprintf("Left-Click: %d",count1++);*

*else if (press==2) cprintf("Right-Click: %d",count2++);*

*}*

1. Vẽ sơ đồ luồng điều khiển cho đoạn chương trình trên?
2. Có tối thiểu bao nhiêu đường bao phủ câu lệnh? Liệt kê?
3. Có tối thiếu bao nhiêu đường bao phủ nhánh? Liệt kê?

**Bài 5:** Cho đoạn chương trình sau:

*puts(s);*

*for(i=0; i<strlen(s)-1; i++)*

*{*

*kt[i].c=s[i];*

*kt[i].f=1;*

*for (int j=i+1; j<strlen(s)-1; j++)*

*if (s[i]==s[j])*

*{*

*kt[i].f++;*

*strcpy(&s[j],&s[j+1]);*

*}*

*}*

1. Vẽ sơ đồ luồng điều khiển cho đoạn chương trình trên.
2. Tính hằng số cyclomatic (kí hiệu C) cho sơ đồ vừa vẽ?
3. Liệt kê C đường độc lập?
4. Sinh tập testcase tương ứng?

**Bài 6:** Cho đoạn chương trình sau:

*puts(s);*

*for (i=0; i<strlen(s)-1; i++)*

*printf("\ntan so xuat hien cua ky tu %c la %d/%d",s[i], kt[i].f,n )*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Có tối thiểu bao nhiêu bao phủ câu lệnh? Liệt kê?
3. Có tối thiểu bao nhiêu bao phủ nhánh? Liệt kê?

**Bài 7:** Cho đoạn chương trình sau:

*void ham1(int \*b, int n)*

*{*

*int i=n;*

*while(i>0 && b[i])*

*{*

*b[i]=0; i--;*

*}*

*if(i==0) stop=1;*

*else b[i]=1;*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Có tối thiểu bao nhiêu bao phủ câu lệnh? Liệt kê?
3. Có tối thiểu bao nhiêu bao phủ nhánh? Liệt kê?

**Bài 8:** Cho đoạn chương trình sau:

*void trave(int \*b,int n)*

*{*

*int i; count++;*

*printf("\nKet qua %d: ",count);*

*for(i=1;i<=n;i++) printf("%3d",b[i]);*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính hằng số Cyclomatic (C)?
3. Liệt kê 1 tập gồm C đường độc lập?

**Bài 9:** Cho đoạn chương trình sau:

*void ham2(int \*b, int n)*

*{*

*int i,stop=0;*

*while(!stop)*

*{*

*trave(b,n);*

*ham1(b,n);*

*}*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính số Cyclomatic?
3. Liệt kê 1 tập gồm C đường độc lập trên?

**Bài 10:** Cho đoạn chương trình sau:

*int length(list L)*

*{*

*int d=0;*

*struct Node \*V=L;*

*while(V!=NULL)*

*{ d++;*

*V=V->link;*

*}*

*return d;*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính số Cyclomatic?
3. Liệt kê tập gồm C đường độc lập trên?

**Bài 11:** Cho đoạn chương trình sau:

*void Search(list L, item x, struct Node \*V)*

*{*

*V=L;*

*while((V!=NULL)&&(V->data!=NULL))*

*V=V->link;*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính số Cyclomatic?
3. Liệt kê tập gồm C đường độc lập trên?

**Bài 12:** Cho đoạn chương trình sau:

*void Insertafter(item x, struct Node\*P, list \*L)*

*{*

*struct Node \*V;*

*V=(struct Node \*)calloc(1,sizeof( struct Node));*

*if(V!=NULL)*

*{ V->data=x;*

*V->link=P->link;*

*P->link=V;*

*}*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính số Cyclomatic?
3. Liệt kê tập gồm C đường độc lập trên?

**Bài 13:** Cho đoạn chương trình sau:

*void Insertbefore(item x,struct Node \*P, list \*L)*

*{*

*struct Node \*Q;*

*Q=\*L;*

*while(Q->link!=P)*

*Q=Q->link;*

*Insertafter(x,Q,L);*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính số Cyclomatic?
3. Liệt kê tập gồm C đường độc lập trên?

**Bài 14:** Cho đoạn chương trình sau:

*void Insertfirst(item x, list \*L)*

*{*

*struct Node\*V;*

*V=((struct Node\*) calloc(1,sizeof(struct Node)));*

*if(V!=NULL)*

*{*

*V->data=x;*

*V->link=\*L;*

*\*L=V;*

*}*

*}*

1. Vẽ sơ đồ luồng cho đoạn chương trình trên?
2. Tính số Cyclomatic?
3. Liệt kê tập gồm C đường độc lập trên?

**d) Hướng dẫn/ Gợi ý**

Xem bài tập mẫu trong slide bài giảng. Nội dung các kỹ thuật kiểm thử hộp trắng trong chương 4 bài giảng.

--------------------------------------------------------------------------------------------------------

## 2.3 Buổi 3: Bài tập kết hợp giữa kiểm thử hộp đen và kiểm thử hộp trắng

**a) Mục tiêu**

Sinh viên nắm vững những kiến thức tổng quan về kiểm chứng phần mềm. Biết vận dụng những kiến thức đã học vào cuộc sống nói chung và thực tế nghề nghiệp nói riêng. Đứng trước một yêu cầu trong thực tế sinh viên có kỹ năng xác định được bài toán, và đưa ra được hướng giải quyết vấn đề.

Giúp sinh viên biết tổng hợp các kỹ thuật đã học để tìm các testcase phù hợp cho từng bài toán.

**b) Nội dung sinh viên phải biết**

- Sinh viên nắm vững và tổng hợp được những kiến thức cơ bản trong lĩnh vực kiểm chứng phần mềm: Các khái niệm; qui trình quản lý.

- Nắm chắc các bước kiểm thử hộp đen

- Các bước tiến hành tạo testcase ứng với từng kỹ thuật: kỹ thuật chia miền tương đương, kỹ thuật test biên,…

- Nắm chắc các bước kiểm thử hộp trắng.

- Các bước tiến hành tạo testcase ứng với từng kỹ thuật: kỹ thuật kiểm thử đồ thị luồng điều khiển, kỹ thuật kiểm thử đồ thị luồng dữ liệu,…

- Và một số test dựa vào kinh nhiệm.

- Áp dụng được vào từng bài tập cụ thể

- Có hiểu biết về xu hướng hiện nay của lĩnh vực kiểm chứng và đảm bảo chất lượng phần mềm.

**c) Nội dung bài tập**

**Bài tập 1:**

Bài toán yêu cầu nhập điểm cho danh sách sinh viên đã tồn tại trong cơ sở dữ liệu. Khi lựa chọn được lớp (tương ứng với mã lớp), mã sinh viên, tên sinh viên hợp lệ, người dùng sẽ có quyền nhập điểm cho sinh viên đó. Thông tin điểm ở đây bao gồm: điểm Toán, điểm Lý và điểm Hóa. Bỏ qua các giá trị đã cố định trong cơ sở dữ liệu là Lớp, Mã sinh viên, Tên sinh viên, chúng ta sẽ tập trung khảo sát và sinh các ca kiểm thử cho các biến đầu vào cần quan tâm là điểm Toán, điểm Lý và điểm Hóa. Với giao diện chương trình như hình 4.1, các biến đầu vào điểm Toán, điểm Lý và điểm Hóa được kí hiệu tương ứng là DToan, DLy, DHoa có miền giá trị vào như sau:

0<= DToan <=10

0<= DLy <=10

0<= DHoa <=10



*Yêu cầu:* Hãy sử dụng các phương pháp đã học test giao diện trên và các chức năng của chương trình trên?

**Bài tập 2:**

Hãy xây dựng hàm NextDate như sau:

Ý tưởng cơ bản của bài toán “NextDate” là cho phép người dùng nhập vào một ngày bất kì (gọi là Current date) và hệ thống sẽ trả về ngày tiếp theo (gọi là Next date) tương ứng với ngày nhập vào. Current date và Next date sẽ được biểu diễn qua 3 biến ngày, tháng, năm (tương ứng Day, Month, Year). Hệ thống yêu cầu các giá trị đầu vào phải đúng định dạng, hợp logic và với giả thiết chỉ khảo sát các năm nằm trong khoảng 1812<= Year <=2012. Như vậy ta xác định được miền giá trị đầu vào của các biến như sau:

1<= Date <=31

1<=Month <=12

12<= Year <=2012

*Yêu cầu:*

1. Xây dựng hàm NextDate
2. Dùng các kỹ thuật đã học xây dựng bộ testcase kiểm tra hàm trên.

**d) Hướng dẫn/ Gợi ý**

Xem bài tập mẫu trong slide bài giảng. Nội dung các kỹ thuật kiểm thử hộp đen, hộp trắng trong chương 4 bài giảng.

--------------------------------------------------------------------------------------------------------

## 2.4 Buổi 4: Thảo luận về công cụ kiểm thử QTpro

**a) Mục tiêu**

Biết cách cài đặt và sử dụng thành thạo công cụ. Biết sử dụng công cụ để test thành công trên một phần mềm cụ thể.

**b) Nội dung sinh viên phải biết**

- Kiến thức tổng quan về kiểm thử tự động.

- Mô hình CMM/ CMMi trong lĩnh vực Công nghệ Phần mềm.

- Hiểu biết về các chức năng chính và sử dụng thành thạo công cụ kiểm thử QuickTest Professional trong công việc kiểm thử phần mềm.

**c) Nội dung chủ đề thảo luận**

Các nhóm trình bày/ thảo luận các vấn đề chính theo yêu cầu của giảng viên bao gồm:

1. Trình bày bản báo cáo kết quả đã tìm hiểu về công cụ theo kế hoạch, bao gồm: Các phiên bản của công cụ; Các chức năng chính; Ưu, nhược điểm; Kết quả so sánh với các công cụ kiểm thử tự động khác.
2. Vận dụng: Sử dụng công cụ để tiến hành test trên một phần mềm hoặc ứng dụng cụ thể.
3. Đưa ra kết quả test.

**d) Hướng dẫn/ Gợi ý**

Đọc nội dung chương 7 trong bài giảng.

--------------------------------------------------------------------------------------------------------

## 2.5 Buổi 5: Thảo luận về công cụ kiểm thử Jmetter

**a) Mục tiêu**

Biết cách cài đặt và sử dụng thành thạo công cụ. Biết sử dụng công cụ để test thành công trên một phần mềm cụ thể.

**b) Nội dung sinh viên phải biết**

- Kiến thức tổng quan về kiểm thử tự động.

- Mô hình CMM/ CMMi trong lĩnh vực Công nghệ Phần mềm.

- Hiểu biết về các chức năng chính và sử dụng thành thạo công cụ kiểm thử Jmetter trong công việc kiểm thử phần mềm.

**c) Nội dung chủ đề thảo luận**

Các nhóm trình bày/ thảo luận các vấn đề chính theo yêu cầu của giảng viên bao gồm:

1. Trình bày bản báo cáo kết quả đã tìm hiểu về công cụ theo kế hoạch, bao gồm: Các phiên bản của công cụ; Các chức năng chính; Ưu, nhược điểm; Kết quả so sánh với các công cụ kiểm thử tự động khác.
2. Vận dụng: Sử dụng công cụ để tiến hành test trên một phần mềm hoặc ứng dụng cụ thể.
3. Đưa ra kết quả test.

**d) Hướng dẫn/ Gợi ý**

Đọc nội dung chương 7 trong bài giảng.

--------------------------------------------------------------------------------------------------------

## 2.6 Buổi 6: Thảo luận về công cụ kiểm thử JUNIT

**a) Mục tiêu**

Biết cách cài đặt và sử dụng thành thạo công cụ. Biết sử dụng công cụ để test thành công trên một phần mềm cụ thể.

**b) Nội dung sinh viên phải biết**

- Kiến thức tổng quan về kiểm thử tự động.

- Mô hình CMM/ CMMi trong lĩnh vực Công nghệ Phần mềm.

- Hiểu biết về các chức năng chính và sử dụng thành thạo công cụ kiểm thử JUNIT trong công việc kiểm thử phần mềm.

**c) Nội dung chủ đề thảo luận**

Các nhóm trình bày/ thảo luận các vấn đề chính theo yêu cầu của giảng viên bao gồm:

1. Trình bày bản báo cáo kết quả đã tìm hiểu về công cụ theo kế hoạch, bao gồm: Các phiên bản của công cụ; Các chức năng chính; Ưu, nhược điểm; Kết quả so sánh với các công cụ kiểm thử tự động khác.
2. Vận dụng: Sử dụng công cụ để tiến hành test trên một phần mềm hoặc ứng dụng cụ thể.
3. Đưa ra kết quả test.

**d) Hướng dẫn/ Gợi ý**

Đọc nội dung chương 7 trong bài giảng.

--------------------------------------------------------------------------------------------------------

# III. Phụ lục

# PHỤ LỤC 1 – CMM, CMMI

**CMMI là gì?**

CMMI viết tắt cho Capability Maturity Model Integration - Mô hình trưởng thành năng lực tích hợp - và là khuôn khổ cho cải tiến qui trình phần mềm. Nó dựa trên khái niệm về các thực hành tốt nhất về kĩ nghệ phần mềm và giải thích kỉ luật mà các công ty có thể dùng để cải tiến các qui trình của họ.

Mô hình CMMI là một khung các giải pháp tối ưu cho quá trình sản xuất phần mềm. Phiên bản CMMI-DEV hiện nay (CMMI cho chuyên viên phát triển), mô tả những giải pháp tốt nhất trong quá trình kiểm soát, đo lường và kiểm tra các quy trình phát triển phần mềm. Mô hình CMMI không tập trung mô tả chính các quá trình mà chỉ mô tả đặc điểm của các quá trình hiệu quả, vì vậy mô hình CMMI đưa ra chỉ dẫn cho các công ty để họ có thể tự mình phát triển hoặc điều chỉnh chính các quá trình của họ.

Mô hình CMMI được mô tả trên trang web chính thức CMMI website: Dự án CMMI là một nỗ lực chung nhằm cung cấp các mô hình để cải thiện nâng cấp các sản phẩm và quy trình. Trọng tâm chính của dự án là tập trung xây dựng các công cụ hỗ trợ việc cải thiện các quy trình dùng để phát triển và ổn định các hệ thống và sản phẩm. Kết quả của dự án CMMI là một bộ các sản phẩm cung cấp một phương pháp tiếp cận tích hợp trên toàn doanh nghiệp để cải thiện các quy trình sản xuất mà vẫn có thể giảm bớt nhân công dư thừa, độ phức tạp, và chi phí từ việc sử dụng các mô hình CMM (quy trình quản lý sản xuất phẩn mềm) riêng lẻ và nhiều mô hình CMM.

**CMMI bắt nguồn từ đâu?**

CMMI là phiên bản kế tiếp của CMM. Cả CMM và CMMI đều được Viện kỹ nghệ phần mềm Mỹ SEI tại trường Đại học Carnegie Mellon ở Pittsburgh, PA phát triển. CMM đã có mặt từ cuối những năm 80 và một thập kỷ sau nó bị CMMI thay thế. Năm 2000 CMMI phiên bản 1.02 được đưa ra thị trường. Phiên bản mới nhất hiện nay CMMI 1.2 được trình làng vào tháng 8 năm 2006. Đôi nét về lịch sử Do cấu trúc của CMMI được thừa hưởng rất nhiều từ CMM, chúng ta hãy xem xét lí do và nguồn gốc để có thể hiểu được cả hai mô hình này có ý nghĩa như thế nào.  
 CMM là kết quả của một nghiên cứu được không quân Mỹ tài trợ, nghiên cứu này được coi là một phương pháp đánh giá khách quan công việc của các nhà thầu phụ về phần mềm. Bộ Quốc Phòng Mỹ cũng quan tâm tới việc chi phí phát triển phần mềm đang leo thang và các vấn đề liên quan đến chất  lượng của các phần mềm nên đã thành lập viện SEI vào đầu những năm 80, và bắt đầu nghiên cứu mô hình CMM vào năm1988.  
 Ban đầu, mô hình CMM được sử dụng như một công cụ để đánh giá khả năng của các nhà thầu chính phủ khi họ tiến hành một dự án phần mềm theo hợp đồng. Mặc dù CMM được thiết kế để đánh giá quá trình phát triển phần mềm nhưng nó đã và đang được áp dụng như một mô hình chung cho kỳ hạn của các quá trình trong các công ty về CNTT hay bất cứ công ty nào khác. Các nhà phê bình cũng nhận thấy CMM luôn được gắn chặt trong một mô hinh phát triển thác nước và không quan tâm tới các khía cạnh khác của quy trình phát triển phẩn mềm như thiết kế và triển khai. CMM không phù hợp với các quy trình ngoại vi liên quan đến việc phát triển phần mềm như là việc mua lại. CMM cũng bị phê phán là tạo ra quá nhiều giấy tờ sổ sách và quá nhiều cuộc họp và nó cũng không phù hợp với nhiều ngành công nghiệp.  
 Các ngành công nghiệp và chính phủ đã tìm ra giải pháp cho vấn đề này bằng cách áp dụng CMM cho các lĩnh vực khác. Toàn bộ quy trình sẽ được giám sát bởi một ban lãnh đạo bao gồm những người đại diện từ OSD, Không quân, Quân đội, Hải quân, các bộ phận khác của chính phủ, SEI và ngành công nghiệp. Nhiệm vụ của ban lãnh đạo này là hướng dẫn và giám sát quá trình phát triển dòng sản phẩm CMMI, đưa ra các sản phẩm CMMI để thẩm tra và phát hành ra công chúng. Viện SEI phối hợp với các chuyên gia đề tài phụ trách quản lý dự án ban đầu là phát triển phần mềm, xây dựng hệ thống, phát triển quy trình và sản phẩm tích hợp. Các cổ đông/ các nhà phê bình đều có quyền kiểm tra, phê bình và đưa ra những gợi ý để phát triển các sản phẩm CMMI. Trong số những người  này cũng có những đại diện từ ngành công nghiệp, chính phủ và viện SEI.

Vậy như ta có thể thấy, CMMI không phải hoàn toàn mới. Hơn thế nữa, CMMI là một sự kết hợp và phù hợp của nhiều biến thể CMM đã phát triển cùng với những yêu cầu của ngành công nghiệp. Hiểu được CMM và nguồn gốc của nó, ta sẽ biết được nền tảng của CMMI. Và CMMI cũng được sử dụng ở hầu hết những nơi giống nhau – theo nghĩa về sự phát triển, không phải là cách mạng. CMMI mang lại sự khôn ngoan của nhiều ngành công nghiệp khác nhau đã giúp CMM phù hợp với những ngành công nghiệpđó.  
**Khác biệt giữa ISO 9001:2000 và CMM/CMMI?**

• ISO 9001 là một tiêu chuẩn quốc tế về quản lý, các điều khoản gọi là “yêu cầu” quy định những điểm cần phải làm (what to do), không chỉ ra việc đó nên làm như thế nào (how to do).

• CMM/CMMi là một mô hình, cung cấp các hướng dẫn và kinh nghiệm thực tế dùng để phát triển, cải tiến và đánh giá năng lực của quy trình.

• CMMi không phải là một tiêu chuẩn, tùy vào từng tổ chức, cách thực hiện khác nhau rất nhiều.

• Về nguyên tắc, ISO bao gồm (ở mức cao) hầu hết các quy trình chủ chốt của CMM/CMMi, tuy nhiên ISO được dùng cho hầu hết mọi ngành nghề, do vậy không cụ thể và gần gũi với công việc đặc thù có liên quan đến phần mềm như CMM/CMMi. ISO không cung cấp các ví dụ và kinh nghiêm cụ thể như CMM/CMMi

**SCAMPI là gì?**

Phương pháp đánh giá chất lượng CMMI tiêu chuẩn để cải tiến quy trình (SCAMPI) cung cấp các phương pháp đánh giá bằng cách sử dụng các mô hình CMMI. Các loại giấy phép của SEI kết hợp với nhau để thực hiện các phương pháp đánh giá SCAMPI và đào tạo những người đánh giá. Có ba mức SCAMPI là: A, B và C. SCAMPI A xem xét các mức độ kỳ hạn và là mức cơ bản để đánh giá trong khi đó mức độ B và C xem xét cách tiếp cận và quá trình triển khai.

**CMMI được dùng để làm gì?**

Các công ty thương mại và chính phủ sử dụng mô hình CMMI để hỗ trợ viêc xác định cải tiến quy trình để xây dựng hệ thống, xây dựng phần mềm và phát triển quy trình và sản phẩm tích hợp.

Công ty sử dụng quy trình này để phát triển, thu thập và duy trì các sản phẩm và dịch vụ và để làm chuẩn cho chính họ chống lại các công ty khác. Các quy trình tốt hơn cũng có thể là những quy trình có giá rẻ hơn và kết quả chất lượng tốt hơn, cũng như là những quy trình này ước tính thời gian thực cho dự án chính xác hơn.   
 Tuy nhiên, cũng giống như tất cả các cơ cấu khác, CMMI không thể nhanh chóng phù hợp với tất cả các công ty mà không ảnh hưởng đến sự phát triển của công ty đó. SEI cho biết việc cải thiện các dự án sẽ được tính bằng tháng và năm chứ không phải chỉ tính bằng ngày và tuần.Vì việc cải thiện dự án thường đòi hỏi phải có nhiều kiến thức và nguồn lực nên các công ty lớn hơn có thể có được kết quả tốt hơn từ CMMI. Tuy nhiên, việc thay đổi quy trình CMMI cũng có thể giúp ích cho các công ty nhỏ hơn. SEI không cấp giấy chứng nhận cho bất cứ loại hình CMMI nào. Đơn giản là SEI chỉ cấp giấy phép hoạt động và cho phép các nhà thẩm định hàng đầu tiến hành đánh giá.

**CMMI Benefits Lợi ích CMMI**

The CMMI Product Suite is at the forefront of process improvement because it provides the latest best practices for product and service development and maintenance. Cải tiến năng lực của các tổ chức phần mềm bằng cách nâng cao kiến thức và kỹ năng của lực lượng lao động. Đảm bảo rằng năng lực phát triển phần mềm là thuộc tính của tổ chức không phải của một vài cá thể. Hướng các động lực của cá nhân với mục tiêu tổ chức. Duy trì tài sản con người, duy trì nguồn nhân lực chủ chốt trong tổ chức.

Lợi ích CMM mang lại cho Doanh nghiệp gói gọn trong 4 từ: Attract, Develop, Motivate và Organize.

Lợi ích CMM mang lại cho người lao động:

- Môi trường làm việc, văn hóa làm việc tốt hơn.

- Vạch rõ vai trò và trách nhiệm của từng vị trí công việc.

- Đánh giá đúng năng lực, công nhận thành tích.

- Chiến lược, chính sách đãi ngộ luôn được quan tâm.

- Có cơ hội thăng tiến.

- Liên tục phát triển các kỹ năng cốt yếu.

**Các level của CMM/CMMI**

CMM bao gồm 5 levels và 18 KPAs (Vùng quy trình quan trọng - Key Process Area). Nói cách khác mỗi một level đều tuân theo một chuẩn ở mức độ cao hơn. Muốn đạt được chuẩn cao hơn thì các chuẩn của các level trước phải thoả mãn. Mỗi level đều có đặc điểm chú ý quan trọng của nó cần các doanh nghiệp phải đáp ứng được.

Level 1 thì không có KPAs nào cả

Level 2 : có 6 KPAs

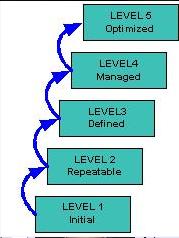
Level 3: có 7 KPAs

Level 4: có 2 KPAs

Level 5: có 3 KPAs

18 KPAs của CMM được đều có 5 thuộc tính(chức năng) chung trong đó có các qui định về key pratice là những hướng dẫn về các thủ tục(**procedure**), qui tắc(**polities**), và hoạt động (**activites**)của từng KPA.

Mô hình này xác định năm cấp độ của CMM đối với một công ty : Khởi đầu (lộn xộn, không theo chuẩn) - Lặp (quản lý dự án, tuân thủ quy trình) - Xác lập (thể chế hóa) - Kiểm soát (định lượng) - Tối ưu (cải tiến quy trình).



**Level 1**

Level 1 là bước khởi đầu của CMM, mọi doanh nghiệp, công ty phần mềm, cá nhóm, cá nhân đều có thể đạt được. Ở lever này CMM chưa yêu cầu bất kỳ tính năng nào. Ví dụ: không yêu cầu quy trình, không yêu cầu con người, miễn là cá nhân, nhóm, doanh nghiệp… đều làm về phầm mềm đều có thể đạt tới CMM này.

***Đặc điểm của mức 1:***

*Hành chính*: Các hoạt động của lực lượng lao động được quan tâm hàng đầu nhưng được thực hiện một cách vỗi vã hấp tấp.

*Không thống nhất:* Đào tạo quản lý nhân lực nhỏ lẻ chủ yếu dựa vào kinh nghiệp cá nhân.

*Quy trách nhiệm*: Người quản lý mong bộ phận nhân sự điều hành và kiểm sóat các hoạt động của lực lượng lao động.

*Quan liêu*: Các hoạt động của lực lượng lao động được đáp ứng ngay mà không cần phân tích ảnh hưởng.

*Doanh số thường xuyên thay đổi*: Nhân viên không trung thành với tổ chức.

**Level 2**

Có 6 KPA nó bao gồm như sau:

- Requirement Management (Lấy yêu cầu khách hàng, quản lý các yêu cầu đó)

- Software Project Planning (Lập các kế hoạch cho dự án)

- Software Project Tracking (Theo dõi kiểm tra tiến độ dự án)

- Software SubContract Managent (Quản trị hợp đồng phụ phần mềm)

- Software Quality Assurance (Đảm bảo chất lượng sản phẩm)

- Software Configuration Management (Quản trị cấu hình sản phẩm => đúng yêu cầu của khách hàng không)

Khi ta áp dụng Level 2, KPA 2(Software Project Planning), ta sẽ có những common feature (đặc điểm đặc trưng) như sau:

*Mục tiêu*(Goal): các hoạt động và những đề xuất của một dự án phần mềm phải được lên kế hoạch và viết tài liệu đầy đủ

*Đề xuất/ Xem xét* (Commitment): dự án phải tuân thủ theo các qui tắc của tổ chức khi hoạch định

*Khả năng*(Ability): Việc thực hiện lập kế hoạch cho dự án phần mềm phải là bước thực hiện từ rất sớm khi dự án đưọc bắt đầu

*Đo lường*(Measument): Sự đo lường luôn được thực thi và sử dụng chúng ta luôn có thể xác định và kiểm soát được tình trạng các hoạt động trong tiến trình thực hiện dự án

*Kiểm chứng*(Verification): Các hoạt động khi lập kế hoạch dự án phải được sự reviewed của cấp senior manager

Để đạt được Level 2 thì người quản lý phải thiết lập được các nguyên tắc cơ bản và quản lý các hoạt động diễn ra. Họ có trách nhiệm quản lý đội ngũ của mình

Các KPA (Key Process Areas) của nó chú trọng tới các thành phần sau :

+ Chế độ đãi ngộ

+ Đào tạo

+ Quản lý thành tích

+ Phân công lao động

+ Thông tin giao tiếp

+ Môi trường làm việc

***Để từ level1 tiến tới level 2 cần có những gì:***

Trước tiên nó phải thỏa mãn các điều kiện ở level1. Tiếp theo là phải chú trọng tới các phần sau:

1. Môi trường làm việc:

- Đảm bảo điều kiện làm việc

- Tạo hứng thú trong công việc

- Không bị ảnh hưởng, mất tập trung bởi các nhân tố khác

2. Thông tin:

Xây dựng cơ chế truyền tin thông suốt từ trên xuống dưới và ngược lại nhằm giúp cá nhân trong tổ chức chia sẽ thông tin, kiến thức, kinh nghiệm, các kỹ năng giao tiếp phối hợp và làm việc hiệu quả

3. Xây dựng đội ngũ nhân viên:

Ngay từ khâu tuyển dụng, lựa chọn kỹ càng và định hướng, thể chế hóa quy trình tuyển dụng

4. Quản lý thành tích:

Đẩy mạnh thành tích, công nhận năng lực, thành tích bằng cách thiết lập các tiêu chí khách quan để đánh giá và liên tục khuyến khích khả năng làm việc, tập trung phát triển sự nghiệp, xây dựng các mục tiêu tiếp theo.

5. Đào tạo:

Không chỉ đào tạo các kiến thức chuyên môn phục vụ cho dự án mà còn mở rộng đào tạo các kỹ năng then chốt, cần thiết như kỹ năng làm việc đội, nhóm, kỹ năng quản lý… nhằm tạo cơ hội cho người lao động phát huy khả năng, cơ hội học hỏi và phát triển bản thân.

6. Chế độ đãi ngộ:

Hoạch định chiến lược đãi ngộ, thu thập ý kiến lực lượng lao động và công bố công khai. Chế độ đãi ngộ cần tập trung vào việc trả lương cho công nhân viên dựa vào vai trò, vị trí của họ (Position), Con người (Person) – thái độ và tác phong làm việc và Thành tích (Performance) mà họ đạt được, cống hiến cho tổ chức. Đưa ra được chính sách lương, thưỏng, phụ cấp các các quyền lợi khác để khuyến khích các cá nhân dựa trên sự đóng góp của họ và cấp độ phát triển của toàn tổ chức.

**Level 3**

Các vùng tiến trình chủ chốt ở mức 3 nhằm vào cả hai vấn đề về dự án và tổ chức, vì một tổ chức (công ty) tạo nên cấu trúc hạ tầng thể chế các quá trình quản lý và sản xuất phần mềm hiệu quả qua tất cả các dự án. Chúng gồm có Tập trung Tiến trình Tổ chức (Organization Process Focus), Phân định Tiến trình Tổ chức (Organization Process Definition), Chương trình Đào tạo (Training Program), Quản trị Phần mềm Tích hợp (Integrated Software Management), Sản xuất Sản phẩm Phần mềm (Software Product Engineering), Phối hợp nhóm (Intergroup Coordination), và Xét duyệt ngang hàng (Peer Reviews).

Để đạt được level 3 thì người quản lý phải biến đổi cải tiến các hoạt động đang diễn ra, cải tiến môi trường làm việc.

Lực lượng lao động sở hữu những kiến thức, kỹ năng cốt lõi

KPA chú trọng tới các yếu tố sau :

+ Văn hóa cá thể

+ Công việc dựa vào kỹ năng

+ Phát triển sự nghiệp

+ Hoạch định nhân sự

+ Phân tích kiến thức và kỹ năng

***Từ Level 2 lên Level 3: Các KPA cần thực hiện***

1. Phân tích kiến thức và kỹ năng:

Xác định những kỹ năng và kiến thức cần thiết để làm nền tảng cho hoạt động nhân sự. Lĩnh vực phân tích này bao gồm: xác định quy trình cần thiết để duy trì năng lực tổ chức, phát triển và duy trì các kỹ năng và kiến thức phục vụ công việc, dự báo nhu cầu kiến thức và kỹ năng trong tương lai.

2. Hoạch định nguồn nhân lực:

Đây là lĩnh vực phối hợp hoạt động nhân sự với nhu cầu hiện tại và trong tương lai ở cả các cấp và toàn tổ chức. Hoạch định nguồn nhân lực có tính chiến lược cùng với quy trình theo dõi chặt chẽ việc tuyển dụng và các hoạt động phát triển kỹ năng sẽ tạo nên thành công trong việc hình thành đội ngũ.

3. Phát triển sự nghiệp:

Tạo điều kiện cho mỗi cá nhân phát triển nghề nghiệp và có cơ hội thăng tiến trong nghề nghiệp, nó bao gồm: thảo luận về lựa chọn nghề nghiệp với mỗi cá nhân, xác định các cơ hội, theo dõi sự tiến bộ trong công việc, được động viên, khuyến khích đạt mục tiêu công việc, giao quyền và khuyến khích thực hiện những mục tiêu trong công việc.

4. Các hoạt động dựa trên năng lực:

Ngoài các kỹ năng, kiến thức cốt lõi còn có hoạch định nhân lực, tuyển dụng dựa vào khả năng làm việc, đánh giá hiệu quả qua mỗi công việc và vị trí, xây dựng chế độ phúc lợi, đãi ngộ dựa trên hiệu quả… giúp bảo đảm rằng mọi hoạt động của tổ chức đều xuất phát từ mục đích phục vụ cho phát triển nguồn nhân lực

5. Văn hóa cá thể:

Tạo lập được cơ chế liên lạc thông suốt, kênh thông tin hiệu quả ở mọi cấp độ trong tổ chức, phối hợp được kinh nghiệm, kiến thức của mỗi người để hỗ trợ lẫn nhau, giúp nhau cùng tiến bộ. Trao quyền thúc đẩy nhân viên tham gia ý kiến, ra quyết định

**Level 4**

Các vùng tiến trình chủ yếu ở mức 4 tập trung vào thiết lập hiểu biết định lượng của cả quá trình sản xuất phần mềm và các sản phẩm phần mềm đang được xây dựng. Đó là Quản lý quá trình định lượng (Quantitative Process Management) và Quản lý chất lượng phần mềm (Software Quality Management). Lực lượng lao động làm việc theo đội, nhóm và được quản lý một cách định lượng.

Các KPA của level 4 chú trọng tới:

+ Chuẩn hóa thành tích trong tổ chức

+ Quản lý năng lực tổ chức

+ Công việc dựa vào cách làm việc theo nhóm

+ Xây dựng đội ngũ chuyên nghiệp

+ Cố vấn

Để đạt được level 4 thì phải đo lường và chuẩn hóa. Đo lường hiệu quả đáp ứng công việc, chuẩn hóac phát triển các kỹ năng, năng lực cốt lõi.

Level 4 này sẽ chú trọng vào những người đứng đầu của một công ty, họ có khả năng quản lý các công việc như thế nào

**Level 5**

Các vùng tiến trình chủ yếu ở mức 5 bao trùm các vấn mà cả tổ chức và dự án phải nhắm tới để thực hiện hoàn thiện quá trình sản xuất phần mềm liên tục, đo đếm được. Đó là Phòng ngừa lỗi (Defect Prevention), Quản trị thay đổi công nghệ (Technology Change Management), và Quản trị thay đổi quá trình (Process Change Management) Để đạt được level 4 thì phải đo lường và chuẩn hóa. Đo lường hiệu quả đáp ứng công việc, chuẩn hóac phát triển các kỹ năng, năng lực cốt lõi.

Để đạt được Level 5 thì doanh nghiệp đó phải liên tục cải tiến hoạt động tổ chức, tìm kiếm các phương pháp đổi mới để nâng cao năng lực làm việc của lực lượng lao động trong tổ chức, hỗ trợ các nhân phát triển sở trường chuyên môn.

Chú trọng vào việc quản lý, phát triển năng lực của nhân viên.

Huấn luyện nhân viên trở thành các chuyên gia.

**Tất cả mọi người đều có thế sử dụng CMMI?**

Vì không phải tất cả các điểm mạnh của CMMI đều phù hợp với tất cả các tổ chức. Với bất cứ cơ cấu hay phương pháp nào thì việc tiến hành CMMI thường thất bại không chỉ do những thiếu sót trong các khái niệm mà còn do việc tiến hành của các tổ chức không được như mong đợi. Vấn đề ở đây có thể là do văn hóa. Ví dụ, CMMI đã từng bị gọi là "chết bởi quy trình". Nếu nhân viên của bạn không được định hướng quy trình đầy đủ thì dường như CMMI sẽ không thể phù hợp nếu không mở rộng đào tạo (và thậm chí còn có thể điều chỉnh một số quan điểm). Cố gắng áp dụngCMMI vào các quy trình đã tồn tại mà không thực hiện phân tích lỗ hổng ban đầu để đánh giá sự phù hợp của CMMI với các quy trình đã tồn tại có phải là một công thức cho sự thất bại hay không. Tương tự như vậy, các nhà phê bình phàn nàn rằng CMMI (cũng giống như CMM trước đây) đòi hỏi người sử dụng phải có một lượng tài liệu khổng lồ. Đây không phải là một trò đùa. Việc đánh giá và tư vấn có thể sẽ rất tốn kém. SEI cho biết một nhóm đánh giá bao gồm từ bốn đến chín thành viên (bạn có thể phải trả cho mỗi người 1000 đô lamột ngày). Và công việc đánh giá này không thể hoàn thành xong trong vòng một hoặc thậm chí hai, ba ngày.

Nhóm đánh giá này không chỉ thảo luận với một vài người hay xem xét một dự án mà họ phải tiến hành kiểm tra nghiêm túc một vài dự án.  
**VIỆT NAM ÁP DỤNG CMM VÀ CMMI TRONG LĨNH VỰC PHẦN MỀM**

Việt nam có các lợi thế về nhân lực cần cù, chăm chỉ, giá nhân công rẻ, hệ thống giáo dục được đào tạo bài bản.... Việt Nam hiện là điểm sáng trên bản đồ thế giới. Tuy nhiên, trong quá trình tiếp thị và giới thiệu tiềm năng với cộng đồng CNTT quốc tế, yêu cầu đầu tiên được phía nuớc ngoài đặt câu hỏi là: Doanh nghiệp CNTT Việt Nam có tốc độ phát triển ra sao ? Năng lực thế nào, và đặc biệt là ứng dụng hệ thống quản lý chất lượng phần mềm đến đâu ?

Về cơ bản, quản lý chất lượng phần mềm là vấn đề không mới, nhưng lại là vấn đề còn yếu của các công ty phần mềm Việt Nam. Một số công ty đã đạt chuẩn quốc tế CMM/CMMI trong nâng cao năng lực và quản lý chất lượng phần mềm, song cũng chỉ gói gọn trong vài công ty gia công cho nước ngoài.

Tuy nhiên, thông thường các công ty phải đầu tư ít nhất 40.000- 50.000 USD cho chi phí tư vấn, đánh giá và khảo sát. Chi phí này thực tế còn cao hơn nhiều sau khi cộng thêm các khoản vé máy bay, ăn ở , đi lại cho các chuyên gia tư vấn, giám sát, đào tạo... Thực tế đây là khoản đầu tư khá lớn đối với các công ty phần mềm VN.

Xét đến thời điểm này, chưa có số liệu chính xác về số lượng doanh nghiệp phần mềm đang áp dụng mô hình đánh giá năng lực sản xuất phần mềm CMM/CMMi tại Việt Nam, nhưng có thể nhắc đến những tên tuổi như PSV (CMMi mức 5: 2005), GCS (CMMi mức 4: 2006), FPT Software (CMM mức 5: 2004) và SilkRoad (CMM mức 3).

Các công ty Việt Nam đua nhau lấy ISO và CMM có hai dạng: 1 là muốn cải tiến quá trình quản lý trong việc phát triển phần mềm, còn 1 là đi theo nhu cầu khách hàng hoặc để quảng cáo.

Với các công ty muốn cải tiến quy trình để phát triển thì việc lấy chứng chỉ hơi lâu vì cần phải đào tạo, chỉ rõ cho tất cả nhân viên thấy được lợi ích của các quy trình này để họ tự nguyện làm theo (vì thật sự thì nó sẽ làm cho nhân viên cảm thấy mình phải làm nhiều hơn, phải lưu giữ đủ thứ giấy tờ, thủ tục,...). Khi tất cả đều nhận thức rõ được vấn đề và thấy rằng ISO hay CMM thật sự sẽ mang lại lợi ích lâu dài cho họ thì lúc đó ISO hay CMM mới góp phần cải tiến quy trình thật sự cho công ty. Dĩ nhiên khi có được ISO hay CMM certification thì công ty cũng đem ra để quảng cáo, để làm lợi thế khi ký hợp đồng. Nhưng thật sự thì ISO hay CMM giúp ích cho công tác quản lý rất nhiều, mọi việc phải được cụ thể hoá thành văn bản chứ không làm việc cảm tính, nói miệng với nhau như trước nữa.

Còn với những công ty lấy ISO hay CMM chỉ vì mục đích quảng cáo thì đâu lại vào đấy thôi vì sau khi đạt được chứng chỉ xong họ lại cất những process đó vào tủ và lại làm việc theo thói quen cũ, chẳng có thay đổi gì hết.

**TỔNG KẾT**

Thứ nhất phải khẳng định CMM là bước phát triển tất yếu của các tổ chức trong thời đại kinh tế tri thức, bởi vì nó là việc kết hợp qui chuẩn và sáng tạo cho cách hoạt động của tổ chức, không cứng nhắc mà linh hoạt thay đổi theo thực tế. Mô hình về tiến hoá của tổ chức khẳng định điều này. Chúng ta cũng hiểu được vì sao bất kì thời đại, nền văn minh nào cũng có thời kì huy hoàng rồi bị sa sút và diệt vong. Mọi tổ chức không thực hiện việc đổi mới, đưa hiểu biết mới của các lớp trẻ vào, tất yếu sẽ bị diệt vong, đây là điều các cấp lãnh đạo cần nhận rõ. Ngày xưa diệt vong của từng triều đại là hàng trăm năm. Ngày nay sự diệt vong của các tổ chức chỉ là hàng chục năm hoặc chưa đến thế.

Nếu đó đã là qui luật chung thì các tổ chức của Việt Nam cũng không ngoại lệ. Do đó càng đưa sớm CMM vào thực tế càng tốt và thúc đẩy sự phát triển ở Việt Nam. Nhưng điều thứ hai cần khẳng định là chỉ có thể thực hiện được CMM nếu đấy là nỗ lực của toàn tổ chức, và trong đó cam kết của lãnh đạo là quyết định. Vì vậy việc huấn luyện về CMM phải là huấn luyện cho toàn tổ chức, không phải chỉ là huấn luyện cho đội ngũ kĩ thuật, tuy rằng ban đầu chúng ta vẫn phải xuất phát từ phía kĩ thuật. Các cấp lãnh đạo, có quyền lực cần được học về CMM theo góc độ bảo đảm sự phát triển tiến hoá của tổ chức.

Và thứ ba, chúng ta cần có được một đội ngũ những người am hiểu về CMM để giới thiệu cho nhiều tổ chức thực hiện. Đội ngũ này phải có khả năng dạy cho mọi loại người trong tổ chức, không chỉ cho các chuyên viên kĩ thuật, người đã sẵn sàng học cái mới. Chúng ta phải đủ khả năng để đối diện với mọi cấp lãnh đạo và cung cấp cho họ những tri thức mới về cách làm việc mới, nhưng phải biết nói theo ngôn ngữ của họ.