Functional requirements

Non-functional requirements

Product constraints

* use case model
* supplementary specification
* glossary

system vs. sofware

sofware-to-be

environment: actors

system boundary

iteration

yêu cầu 🡪 phân tích thiết kế 🡪 cài đặt 🡪 kiểm thử

use case – driven

vdụ: hthống đkmh

Ai tương tác với hthống?

Tương tác làm gì? Mục tiêu? (là mục đích)

Tác nhân tương tác ntn? Hthống phản hồi ntn? Được đặc tả trong đặc tả ca sử dụng (use-case specification)

Các tương tác gián tiếp với hệ thống. statement

Vdụ: đkhm. Khoa mong muốn chọn học phần phù hợp…

Concepts- abstraction: cơ sở object/class.

Requirements engineering 🡪 cơ sở để xác định requirement

Lựa chọn topic:

System as is: hiện trạng của hệ thống 🡪 giải pháp

Use-case: mô tả quá trình tương tác giữa tác nhân và hệ thống 🡪 kết quả là mục đích đạt được của tác nhân.

2 loại actor: primary actor – secondary actor

Primary actor: chủ động khởi tạo use case để đạt được goal

Secondary actor: giúp hoàn thành use case được khởi tạo.

Vdụ face plus

primary actor giảng viên create use case điểm danh 🡪 goal: điểm danh lớp học

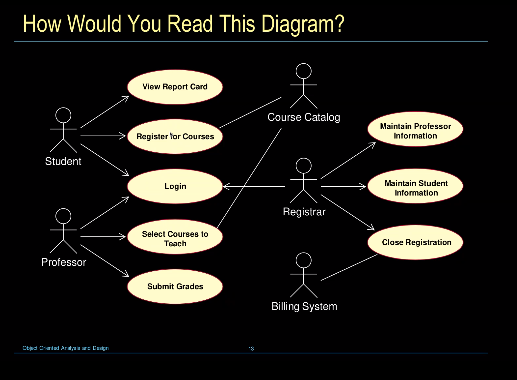
sub –goal: lấy danh sách lớp học phần

second actor: hệ thống QLĐT.

Use-case: chuỗi tương tác giữa actor – system end 🡪 giá trị quan sát được cho tác nhân

Benefits usecase:

* 1
* 2
* 3



Couse catalog là actor. Why? Course catalog là component - software có sẵn , được tái sử dụng

Db k là actor.

Use case: thanh toán lương:

Cuối tháng tự động kích hoạt.

Tác nhân? Clock

Đặc tả ca sử dụng

Scenario – kịch bản: là một đường thực thi ca sử dụng – là 1 luồng (flow of events)

Use-case slice – 1 kịch bản/story + test case.

Iteration = use case slice

Xác định: use case – tách thành các use case slice

Activity diagram.

System sequence diagram (SSD): miêu tả tương tác 2 đối tượng: system – actor

Domain concepts

Checkpoints:

Use case hiểu hết chưa?

Phân chia use case thành packages hợp lý?

….