**Software Engineering Q/A Sheet (#6)**

date: 2023/10/26 number:2018312280 name: 이상수

**Questions from Prof.**

1. Describe the purpose of system modeling of existing and new systems, respectively.

시스템 모델링은 어떠한 관점에 대해 특정 프로세스를 추상화된 모델로 생성하는 과정이며, 오늘날의 시스템 모델링은 그래픽을 활용하여 나타내는 방법을 주로 뜻한다. 이러한 시스템 모델을 활용하면, 분석가 입장에선 시스템의 기능을 이해하는데 도움이 되고, 고객에게 고객이 원하는 모습이 맞는지 확인하는 소통의 도구가 된다.

Models of the existing system :

존재하는 시스템의 모델링에 대해서는, requirement engineering 과정에서 실행된다.

모델링은 존재하는 시스템의 동작을 이해하는데 도움을 주고,

이 시스템의 강점과 약점을 파악하게 해준다.

또한 이 모델은 새 시스템의 요구사항으로 이어진다.

Models of the new system :

새 시스템의 모델링은, requirement engineering과정에서 만들어져서 stakeholder들에게 요구사항을 이해시키는데 도움을 준다. 또한 개발자들은 이 모델을 써서 디자인을 하는데 활용하고, 시스템 개발에 필요한 문서로 활용한다.

1. Explain the four perspectives of system modeling and connect the appropriate diagrams in UML to each perspective. (External, Interaction, Structural, Behaviour perspective)

External perspective :

시스템의 외부적 관점, 환경 등을 모델링한다.

Activity diagram

Interaction perspective :

시스템과 시스템 사이의 환경 사이, 그리고 구성요소들 간의 interaction을 모델링한다.

Use case diagram, sequence diagram 등

Structural perspective :

시스템이 어떻게 구성 되냐, 구조가 어떻게 되느냐를 모델링 하는 것이다.

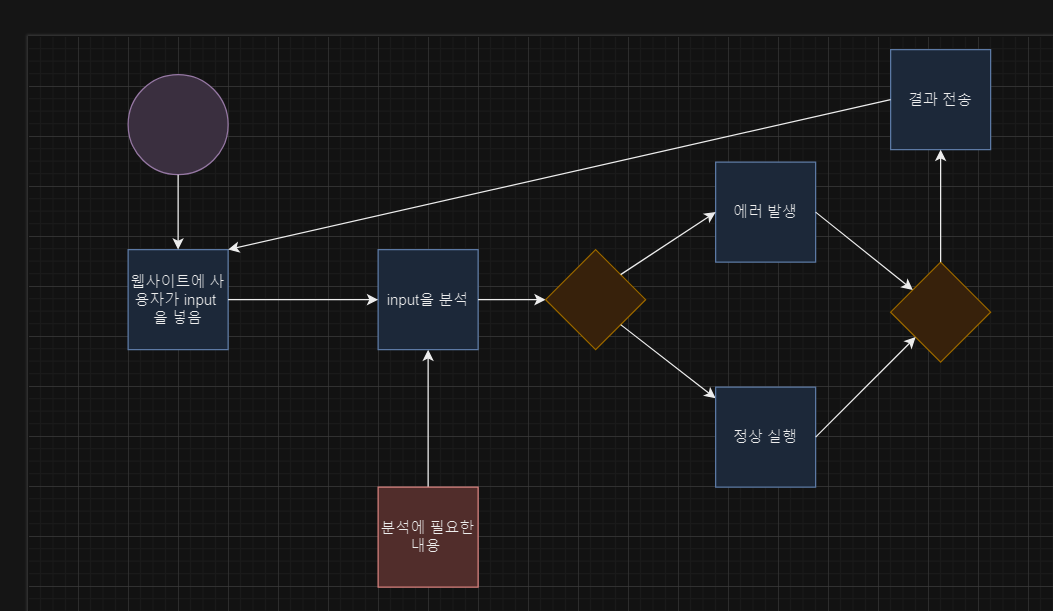
Class diagram

Behavioral perspective :

시스템의 행동과 반응을 모델링하는 것이다.

State diagram

1. Draw a process diagram with a specific example for your term project.



1. Draw use case diagrams with a specific example for your term project.

스크린샷, 블랙, 도표, 라인이(가) 표시된 사진

자동 생성된 설명

Actor : 탄소발생량 계산기, 유저

Description : 탄소발생량 계산 시스템이 계산을 완료하면 결과를 유저에게 보내준다.

Data : 탄소발생량 정보

Stimulus : 유저의 ‘발생량 계산’ 버튼 클릭

Response : 탄소발생량 정보를 알려줌

Comments : 유저는 오류가 없는 코드를 써야하고, 계산기는 악성 코드를 방지해야한다.

1. Learn and explain more about component, package, and deployment diagram.   
   package diagram은 그룹들의 서로 간에 연관성을 나타낸다. Class, object등

Component diagram 은 어떻게 시스템을 여러 파트로 나눌지, 파트끼리 연결을 어떻게 할지 등 시스템 구조를 나타낸다.

Deployment diagram 은 각 소프트웨어 구조가 어떻게 하드웨어에 매핑 될지를 나타낸다.

1. Explain why generalization is effective for complexity management.

Generalization 을 하고나면, 상위 클래스가 밑에 클래스들의 components들을 가지고 있을 것이기에, 수정이 필요하거나 해야 할 때, 상위 클래스 하나만 수정하여도 밑에 클래스들이 같이 수정되기에 개발이 간편해진다.

1. Compare data-centric and event-centric models for purpose of use and modeling techniques.

data-centric models의 경우, 데이터의 입력에 의해 출력이 일어난다. 입력되는 데이터를 처리하고, output을 출력해야한다 는 것을 모델링 하는 것. Activity model을 활용하여 각 data를 받으면 일어나는 activities를 연결하여 나타낸다.

Event-centric models는 외부, 내부의 이벤트에 따라 시스템의 반응을 나타낸다. 시스템이 한정된 state와 event를 가지고 있다고 가정한다.

Statechart, state diagram을 쓴다. 각 state-activity를 연결한 노드들을 event로 연결한다.

1. Explain the advantages and disadvantages of model-driven engineering.

시스템의 추상화 레벨을 올려준다. 모델링을 하면 코드를 수정하면 translator가 자동으로 플랫폼에 맞게 변환을 시켜준다.

어차피 추상화된 내용은 다시 사람의 손으로 수정해서 시스템에 적용되어야한다.

MD(model-driven engineering)을 통해 모델링 하는 비용이 클 수 있다.

1. Compare CIM, PIM, and PSM in MDA.

CIM : computation independent model은 도메인의 추상화를 실행한다.

PIM : platform independent model 은 시스템의 동작을 나타낸다. 흔히 UML모델을 활용해 static system structure이나 how it response to external and internal events를 나타낸다.

PSM : platform specific models는 PIM의 변형으로 각 platform별로 존재하여 각 platform에 특수한 설명을 나타낸다.

1. Explain the slide 49 about MDA transformation flow.

도메인 전문가가 작성한 CIM과 platform에 전문가가 작성한 PSM을 활용하여 개발자는 PIM에만 집중하며 개발한다. 중간에 Translator는 각 모델을 서로 합쳐지게 해주며 변환을 해주는 작업을하여, 최종 적으로 executable code를 만들어낸다.

Questions from your ownself

1. CIM, PSM이 전문가들이 작성을 한다고 하면, requirement engineering에서 domain에 대한 조사를 할 필요가 있었던 것일까?

CIM, PSM에서 부족한 부분들, 또는 개발에 도움이 될 지도 모르기에 그래도 domain에 대한 조사를 진행하는 것이 유용하다.