소프트웨어공학

유스케이스 시나리오 작성

유스케이스 시나리오 작성 (유스케이스 별 내부 모델링)

- 유스케이스 시나리오(기술서, 명세서, 설명서) 작성 ☞ 유스케이스 다이어그램을 보완, 유스케이스 별로 작성, 상세 기능의 흐름

  - 액터와 유스케이스 간의 상호작용을 기반으로 순차적 시나리오 형식으로 기술한다.

  - 시스템이 해당 기능을 수행하기 위해 어떤 절차를 거치는지, 그 과정에서 주고 받는 정도 파악 가능

  - 시스템 개발 전체 단계에서 다양한 이해관계자들과 관련되므로 이해성 및 가독성을 높일 수 있도록 작성한다.

  - 요구사항 분석 단계에서 작성하기 때문에 (사용자 관점에서 기술하며) 액터와 시스템이 상호작용하는 과정을 구체적으로 묘사한다.

  - 시나리오 문장은 '액터' 또는 '시스템'으로 시작해서 명사 뒤에 행위를 나타내는 동사를 기술하여 '주어- 목적어- 동사'의 일관된 문장으로 작성하는 것이 좋다.

    . '액터는 ~을 한다' 형식 (예: 고객은 조회하고자 하는 카테고리를 선택한다. (입력))

    . '시스템은 ~을 한다' 형식 (예: 시스템은 선택된 카테고리의 하위 카테고리를 출력한다.(출력))

유스케이스 시나리오 작성

- 유스케이스 기본 정보 (**정형화된 방법은 없으나 보편적으로 다음과 같은 내용을 포함**)
  - 유스케이스 이름과 개요
  - 유스케이스를 수행하는 행위자(액터)
  - 유스케이스 내용
  - 상태(Status) : 유스케이스 작성 진행 단계를 기술 (예: 작성중)
  - 사전 조건(Preconditions) : 유스케이스의 기본 흐름이 올바르게 동작되기 위하여 유스케이스를 시작하기 전에 만족해야하는 조건을 제시

    (예: 학생은 재학중이고 로그인 되어있어야 한다)
  - 사후 조건(Postconditions) : 유스케이스 종료 이후 정상적인 시스템 상태를 기술

    (예: 수강 신청된 강좌가 스케줄로 저장된다.)
  - 이벤트 흐름(기본 흐름, 대안 흐름, 예외 흐름)

유스케이스 시나리오 작성

- 유스케이스 흐름 정보

  - 기본 흐름(Basic Flows) : 유스케이스를 시작해서 종료할 때까지 정상적으로 발생하는 상호작용의 흐름을 계층화, 구조화하고 번호를 부여하여 단계별로(순차적으로) 기술, 조건문 반복문 등 포함 가능

  - 대안 흐름(Alternative Flows) : 유스케이스의 정상적인 흐름에 대한 대안을 제시하는 것으로 각 단계에서 조건이 false일 때 기본 흐름으로 분기 (필수는 아님) , 기본 흐름과 연계되어 발생

    특정 시점에 여러가지 선택적인 흐름으로 나뉘어질 경우 발생하는 흐름

    ☞ 번호는 기본 흐름 번호와 함께 1a 등의 형식으로 작성

  - 예외 흐름(Exceptional Flows) : 유스케이스 각 단계에서 비정상적인 조건(예외 상황이나 오류)이 발생할 때 기술 (필수는 아님), 기본 흐름과 연계되어 발생

    ☞ 번호는 기본 흐름 번호와 함께 1b 등의 형식으로 작성

※ 기본흐름과 대체 흐름의 2가지 단계로 나누기도 한다.

유스케이스 흐름 작성 예시

흐름

1. 기본 흐름

   1.1 사용자는 ~을 한다. (입력) -> 구체적으로 작성

   1.2 시스템은 ~을 한다. (출력) -> 구제척으로 작성

   1.3 사용자는 ~을 한다. (A1)

   1.4 시스템은 ~을 한다. (출력)

   1.5 사용자는 ~을 한다. (입력)

   1.6 시스템은 ~을 한다. (E1)

   :

2. 대안 흐름

   A1. ~일 경우에는, 시스템이 ~을 한다.

3. 예외 흐름

   E1. ~일 경우에는, 시스템이 ~을 한다.

예시

- ATM 기기에서 돈을 찾는 유스케이스 시나리오 (쉽게 작성하고, 쉽게 이해할 수 있는 기술서)

  - 내가 ATM 기기 앞에서 서서 실제 돈을 찾는 것과 같은 상황을 기술, 해당 기능 요구 사항의 절차(과정)을 기술

1. UC_0001: Withdraw Cash
   - 1.1 설명고객이 ATM으로부터 자신의 계좌에 입금되어 있는 현금을 출금한다.
   - 1.2 이벤트흐름
     - 1.2.1 기본흐름
       1. 사용자는 ATM 기계에 카드를 넣는다.
       2. ATM은 사용자에게 초기 화면을 출력한다.
       3. 사용자는 ATM기계의 인출 버튼을 클릭한다.
       4. ATM은 인출 화면을 출력한다.
       5. 사용자는 ATM에 인출금액을 입력하고 확인 버튼을 클릭한다. (A1)
       6. ATM은 사용자에게 영수증을 발급여부를 물어본다. (E1)
     - 1.2.2 대안 흐름
       A1. 통장 잔액이 부족한 경우, 인출 불가 화면을 띄운다.
     - 1.2.3 예외 흐름
       E1. 영수중 인쇄지가 부족한 경우, 시스템 에러 화면을 띄운다.

예시

A인지 E인지는 내가 결정

- **개요**
  - 학생이 금번학기에 개설된 강좌 중 수강하고자 하는 과목을 신청한다.
- **흐름**
  - 기본 흐름 (Basic Flow)
    1. 학생이 수강신청하기 메뉴를 요청함으로써 이 Use Case는 시작한다.
    2. 학생은 "조회하기" 버튼을 클릭하여 수강 정보를 요청한다.(E5)
    3. 금번학기에 개설된 강의 정보(기신청학점수, 학기, 강좌번호, 강좌 이름, 담당학과, 학점, 강좌 설명, 신청가능여부, 수강신청여부)를 출력한다. (A1, E1)
    4. 학생이 수강신청 하고자 하는 수강과목을 선택한다. (E2,E3,E4)
    5. 선택된 강의정보를 "수강신청하기" 버튼을 클릭하여 신청한다. (E3,E5)
    6. "***건[성공] *건[실패] 로 수강과목 신청이 처리되었습니다.**"라는 메시지를 출력한다.
    7. 신청된 수강과목 목록을 출력한다.
  - 대안 흐름(Alternative Flow)
    - A1. 수강 신청 기간이 아닌 경우, "수강신청 기간이 아닙니다" 라는 메시지를 출력한다.

예시

- **예외 흐름(Exception Flow)**
  - E1 : 검색 실패
    - » 금번학기 수강신청 가능한 개설된 강의목록이 비어있는 경우 등록된 에러 메시지를 출력하고ERR_FFD_NO_CLASS에러를 반환한다.
  - E2 : 최대수강신청 학점 초과 실패
    - » 학생이 최대신청 수강학점을 초과하였을 경우 등록된 에러 메시지를 출력하고 ERR_FFD_OVERCLASS에러를 반환한다.
  - E3 : 최대수강신청인원 초과 실패
    - » 최대수강신청인원 초과하였을 경우 등록된 에러 메시지를 출력하고 ERR_FFD_MAXCLASS에러를 반환한다.
  - E4 : 수강신청 중복 오류 실패
    - » 중복 수강신청이 되었을 경우 등록된 에러 메시지를 출력하고 ERR_FFD_DUPCLASS에러를 반환한다
  - E5 : 강의정보신청 오류 실패
    - »강의정보 신청 시 시스템 오류가 발생되었을 경우 등록된 에러 메시지를 출력하고 ERR_FFD_SYSTEM에러를 반환한다.

유스케이스 모델링 완료

- 유스케이스 시나리오까지 작성되면 유스케이스 모델링이 완료되었다고 볼 수 있다.
- 유스케이스는 사용자가 시스템을 활용할 때 나타나는 쓰임새 혹은 시스템이 사용자에게 제공하는 서비스를 나타낸 것으로써, 행위자가 자신의 목적을 위해서 시스템에서 제공받는 가장 단순하고 직관적인 기능이다.
- 행위자가 시스템에게 요구하는 기능들이 유스케이스로 표현되기 때문에 유스케이스는 시스템의 최상위 요구사항이라고도 볼 수 있다.
- 유스케이스 모델링은 시스템을 블랙박스로 보고 사용자의 관점에서 시스템을 분석한 것
- 유스케이스 식별부터 유스케이스 시나리오 작성까지의 전 과정을 살펴보면 우선 유스케이스 식별은 시스템의 최상위 기능을 찾아내는 것으로서 초기 기능 모델링이 이루어진 것
- 유스케이스 시나리오는 행위자와 시스템 사이에 이루어지는 유스케이스의 진행 과정과 흐름을 나타낸 것으로서 시스템에 대한 동적 모델링을 수행한 것
- 또한 유스케이스 시나리오에는 이벤트 흐름과 함께 행위자와 시스템이 주고받는 정보들이 함께 나타나게 되어, 시스템 내부에서 저장되고 관리되어야 하는 중요 정보들에 대한 초기 정보 모델링이 이루어진 것

Enterprise Architect 실습

- 실습은 수업 내용 참고

T h a n k y o u

TECHNOLOGY

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Velit ex
ullicabo ipsum, labore sed tempora ratione asperiores des
cipserat bore sed tempora rati jgert one bore sed tem!