

Session027 시스템 인터페이스 요구사항 분석

1 시스템 인터페이스 요구사항 구성

- 1.1 시스템 인터페이스 : 독립적으로 떨어져 있는 시스템들끼리 서로 연동하여 사용하기 위한 접속 방법이나 규칙
 - 시스템 인터페이스 요구사항 : 시스템과 외부 시스템을 연동하는데 필요한 시스템 인터페이스에 대한 요구사항을 기술한 것
 - 명세서에는 인터페이스 이름, 연계 대상 시스템, 연계 범위 및 내용, 연계 방식, 송신 데이터, 인터페이스 주기 (수신 데이터 포함x)

2 시스템 인터페이스 요구사항 분석

- 2.1 명세서에서 요구사항을 기능 비기능으로 분류, 조직화하여 요구사항 명세를 구체화하고 이해관계자에게 전달하는 과정
- 2.2 요구사항 분석 기법으로 적절히 이용
- 2.3 필요한 경우 적절한 수준으로 분해 세분화
- 2.4 분석 시 누락된 요구사항이나 제한조건을 추가
- 2.5 상대적 중요도 평가 우선순위 부여

3 시스템 인터페이스 요구사항 분석 절차

- 요구사항 목록에서 시스템 인터페이스 관련 요구사항 선별
- 관련 자료 준비
- 기능 비기능 분류
- 분석 및 명세서 구체화(내용 추가 또는 수정)
- 이해관계자에게 전달 (명세서 공유)

Session028 인터페이스 요구사항 검증

1 요구사항 검증(Requirements Verification)

- 1.1 유저들의 요구사항이 명세서에 정확하고 완전하게 기술되었는지 검토, 개발 범위의 기준인 베이스라인 설정
 - 설계 및 구현 중 명세서 오류 발견은 수정에 많은 비용이 소요
 - 계획수립 -> 검토 및 오류 수정 -> 베이스라인 설정

2 인터페이스 요구사항 검토 계획 수립

- 2.1 이해관계자들이 프로젝트 품질 관리 계획 참조해 계획수립
 - 검토 기준 및 방법 : 프로젝트 규모, 참여 인력, 검토 기간 등 고려
 - 참여자 : 이해관계자, 프로젝트 관리자, 품질 관리자, 인터페이스 분석가, 소프트웨어 아키텍트, 시스템 사용자, 테스트 관리자
 - 체크리스트
 - 관련 자료 : 요구사항 목록, 요구사항 명세서, 현행 및 표준 시스템 구성도 등
 - 일정

- 2.2 참여자들에게 검토 관련 자료와 일정 등을 전달
- 3 인터페이스 요구사항 검토 및 오류 수정
 - 3.1 검토 시 오류 발견 시, 오류 목록과 시정 조치서 작성
 - 3.2 오류 수정 및 요구사항 승인 절차를 위해 검토 결과를 검토 관련자들에게 전달
 - 3.3 시정 조치가 완료 되었는지 확인, 완료되면 검토 작업 완료
- 4 인터페이스 요구사항 베이스라인 설정
 - 4.1 검증된 요구사항은 관리자와 주요 의사 결정자에게 공식적 승인을 받는다
 - 설계 및 구현을 위해 명세서의 베이스라인을 설정
 - 설정 후 요구사항 변경은 공식 절차로만
- 5 요구사항 검증 방법
 - 5.1 요구사항 검토 : 결함 여부를 검토 담당자들이 수작업으로 분석하는 방법
 - 동료검토(Peer Review) : 작성자가 내용을 직접 설명, 동료가 들으면서 결함 발견하는 형태
 - 워크스루(Walk Through) : 회의 전 명세서 미리 배포, 사전 검토 후 짧은 회의를(1, 2 시간) 통해 결함 발견하는 형태
 - 인스펙션(Inspection) : 작성자를 제외한 다른 전문가들이 확인하면 결함 발견하는 형태
 - 5.2 프로토타이핑(Prototyping) : 견본품을 만들어 최종 결과물을 예측
 - 5.3 테스트 설계 : 요구사항은 테스트할 수 있도록 작성되어야 한다. 이를 위해 Test Cast를 생성. 이후 현실적으로 테스트 가능한지 검토
 - 5.4 CASE(Computer Aided Software Engineering) 도구 활용 : 요구사항 변경사항의 추적 및 분석, 관리, 표준 준수 여부를 확인
- 6 인터페이스 요구사항 검증의 주요 항목
 - 완전성(Completeness) : 누락되지 않고 완전하게 반영 되었는지
 - 일관성(Consistency) : 모순되거나 충돌되는 점은 없는가
 - 명확성(Unambiguity) : 모든 참여자가 명확히 이해할 수 있는가
 - 기능성(Functionality) : '무엇을'에 중점을 두었는가
 - 검증 가능성(Verifiability) : 검증할 수 있는가
 - 추적 가능성(Traceability) : 설계서를 추적할 수 있는가
 - 변경 용이성(Easily Changeable) : 변경이 쉽도록 작성되었는가

Session031 인터페이스 방법 명세화

- 1 인터페이스 방법 명세화의 개념
 - 1.1 연계하여 작동할 때 인터페이스별 송수신 방법, 송수신 데이터, 오류 식별 및 처리 방안을 문서로 명확하게 정리하는 것

- 송수신 방법 명세화를 위해서는 시스템 연계 기술, 인터페이스 통신 유형, 처리 유형, 발생 주기 등에 대한 정보 필요
- 2 시스템 연계 기술
 - 2.1 개발할 시스템과 내외부 시스템을 연계할 때 사용되는 기술
 - 2.2 DB Link, API/Open API, 연계 솔루션, Socket, Web Service 등
 - DB Link : DB에서 제공하는 객체를 이용하는 방식
 - API/Open API : 송신 시스템의 DB에서 데이터를 읽어 와 제공
 - 연계 솔루션 : EAI서버와 송수신 시스템에 설치되는 클라이언트를 이용
 - Socket : 소켓을 생성하여 포트 할당, 통신 요청 시 클라이언트와 연결하여 통신하는 네트워크 기술
 - Web Service : WSDL, UDDI, SOAP 프로토콜을 이용하는 서비스
 - 3 인터페이스 통신 유형
 - 3.1 데이터를 송수신하는 형태
 - 단방향 : 시스템에서 거래를 요청만 하고 응답이 없는 방식
 - 동기 : 거래를 요청하고 응답이 올 때까지 대기(Request-Reply)
 - 비동기 : 거래를 요청하고 응답 올 때까지 다른 작업 수행, 응답이 오면 처리
 - 4 인터페이스 처리 유형
 - 4.1 송수신 데이터를 어떤 형태로 처리할 것인지
 - 실시간 방식 : 요청을 바로 처리해야 할 때
 - 지연 처리 방식 : 매건 단위로 처리할 경우 비용이 클 경우
 - 배치 방식 : 대량의 데이터를 처리할 때
 - 5 인터페이스 발생 주기
 - 5.1 송수신 데이터가 전송되어 인터페이스가 사용되는 주기
 - 5.2 업무의 성격과 데이터 전송량을 고려 매일, 수시, 주 1회 등으로 구분
 - 6 송 수신 방법 명세화
 - 6.1 연계 방식, 통신 유형, 처리 유형, 발생 주기 등의 송수신 방법을 정의 명세를 작성
 - 6.2 아키텍처 정의서를 기반으로 하여 업무 및 데이터의 성격, 연계 데이터 발생 건수, 연계 시스템의 기술 구조, 시스템 간의 성능 고려

인터페이스 송·수신 방법 명세화

인터페이스 ID	인터페이스명	송신 시스템	수신 시스템	연계 방식	통신 유형	연계 처리 형태	연계 주기
IFID-001	지급 정보 전송	회계	길벗은행	EAI	요청/응답	실시간	매일
IFID-002	은행 수금 내역 수신	회계	길벗은행	EAI	요청/응답	실시간	매일
IFID-003	은행 계좌 잔액 수신	회계	길벗은행	Socket	요청/응답	실시간	수시
IFID-004	예금주 조회	회계	길벗은행	Socket	요청/응답	실시간	수시
IFID-005	법인 카드 사용 내역 조회	회계	길벗카드	Web Service	요청/응답	실시간	매일
IFID-006	고객 로그인	홈페이지	고객	EAI	단방향	배치	매일
IFID-007	고객 계약 조회	홈페이지	고객	DB Link	단방향	배치	매일

7 송 수신 데이터 명세화

7.1 각각의 인터페이스에 대해 인터페이스 시 필요한 송수신 데이터에 대한 명세 작성

송·수신 데이터 명세화

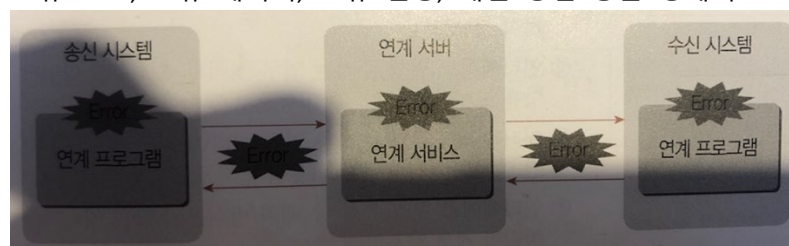
인터페이스 ID	IFID-006
송·수신 구분	수신
시스템명	고객 시스템
업무	고객 정보 관리
서비스명	고객 정보 저장

데이터 항목							
Seq	Field*	Key*	Type	Size*	Null 허용	Description*	Condition*
1	CON_NO	Y	varchar	13	N	고객번호	
2	REQ_CNT	Y	varchar	2	N	반복회차	
3	SEQ		varchar	2	N	순번	
4	SSN		char	13	N	주민번호	암호화*
5	NAME		char	10	N	이름	암호화
6	LICN_CODE		char	16	N	자격코드	코드

8 오류 식별 및 처리 방안 명세화

8.1 인터페이스 시 발생할 수 있는 오류 식별, 오류 처리 방안에 대한 명세서를 작성하는 것

- 시스템 연계 시 발생할 수 있는 다양한 오류 상황을 식별하고 분류
- 오류 코드, 오류 메시지, 오류 설명, 해결 방법 등을 명세화



Session 033 미들웨어 솔루션 명세

1 미들웨어(Middleware)의 개념 및 종류

- 운영체제와 응용프로그램 사이에 추가적인 서비스를 제공하는 소프트웨어
- 표준화된 인터페이스 제공, 데이터 교환에 일관성을 보장

2 DB(Database)

- 클라이언트에서 데이터베이스와 연결하기 위한 미들웨어
- ODBC, IDAPI, Glue
- 3 RPC(Remote Procedure Call)
 - 응용 프로그램의 프로시저를 사용, 원격 프로시저를 로컬 프로시저처럼 호출하는 미들웨어
 - Entera, ONC/RPC
- 4 MOM(Message Oriented Middleware)
 - 메시지 기반의 비동기형 메시지를 전달하는 방식의 미들웨어
 - 이기종 분산 데이터 시스템의 데이터 동기를 위해 많이 사용
 - MQ, Message Q, JMS
- 5 TP-Monitor(Transaction Processing Monitor)
 - 예약 업무 등과 같은 온라인 트랜잭션 업무에서 트랜잭션을 처리 및 감시하는 미들웨어
 - 빠른 응답 속도 유지해야 하는 업무에 사용
 - tuxedo, tmax
- 6 ORB(Object Request Broker)
 - 객체 지향 미들웨어 코바(CORBA) 표준 스펙을 구현
 - 트랜잭션 처리 및 모니터링 등을 추가로 구현
 - Orbix, CORBA
- 7 WAS(Web Application Server)
 - 사용자의 요구에 따라 변하는 동적인 콘텐츠를 처리하기 위해 사용
 - 클라이언트/서버 환경보다는 웹 환경 구현 위한
 - HTTP 세션 처리, Mission-Critical 한 기업 업무까지 JAVA, EJB 컴포넌트 기반으로 구현 가능
 - WebLogic, WebSphere
- 8 미들웨어 솔루션 식별
 - 개발 및 운영 환경에 사용될 미들웨어 솔루션을 확인, 목록 작성
 - 시스템, 구분, 솔루션명, 버전, 제조사 등의 정보를 정리한 목록 작성
 - 이해관계장 등에게 전달, 오류 및 누락을 확인 및 수정
- 9 미들웨어 솔루션 명세서 작성
 - 솔루션별로 관련 정보들을 상세하게 기술하는 것
 - 솔루션에 대한 제품안내서 및 설명 자료 등을 통해 검토
 - 사용 환경, 특징 등을 솔루션 설명 자료나 관련 담당자를 통해 검토
 - 지원하는 시스템 범위와 환경 구성, 제공 기능 등에 대한 제약사항을 제품안내서 및 기술 지원 담당자를 통해 검토
 - 상세정보 및 제공 기능, 특징, 시스템 구성 환경 등에 대한 제약사항을 정리, 솔루션에 대한 명세서를 작성