2021년 05월 15일 기출문제

1. 시스템의 구성요소

* 입력, 처리, 출력, 제어, 피드백

1. 유스케이스

* 시스템이 액터에게 제공해야 하는 기능, 시스템의 요구사항이자 기능
* 유스케이스 다이어그램은 사용자의 **요구를 추출하고 분석하기위해** 주로 사용한다.
* 사용자 액터: 시스템이 제공해야하는 기능인 유스케이스의 권한을 가지는 대상
* 시스템 액터: 사용자 액터가 사용한 유스케이스를 처리해주는 외부 시스템

1. 도출 분석 명세 검토

6. GOF 디자인 패턴

- bridge pattern은 구현부에서 추상층을 분리하여 각자 독립적으로 확장이 가능하게 하는 패턴

- factory method pattern은 상위 클래스에서 객체를 생성하는 인터페이스를 정의하고 하위클래스에서 인스턴스 생성하는 방식

- prototype pattern은 protoype 먼저 생성하여 인스턴스를 복제하여 사용

- mediator은 객체간의 통제와 지시의 역할을 하는 중재자를 주어 객체지향의 목표를 달성하게 해줌

-Adaptor pattern 은 기존의 구현되어 있는 클래스에 기능 발생시 기존 클래스를 재사용

9. 연계시스템

- 송신 시스템: 연계할 데이터를 DB와~~ 송신

- 수신 시스템: 관리하는 데이터 형식에 맞춰 저장하거나 활용하도록 제공

- 중계 서버: 송/수신 시스템사이에서 데이터를 송수신하고 연계데이터를 모니터링함, 연계데이터 보안강화

10. CASE

- 구조적기법, 프로토타이핑기술, 자동프로그래밍 기술, 정보저장소기술, 분산 처리기술

12. 아키텍처 스타일

- 클라이언트 서버 구조: 컴포넌트가 다른 컴포넌트에게 서비스를 요청 , 여러 컴포넌트를 거치며처리

- 계층구조: 모듈들로 응집된 계층 단위로 SW구성, 계층간에 사용가능의 관계로 표현

- MVC구조: 모델-뷰-컨트롤러로 기능을 분리한 아키텍쳐

- 파이프 필터: 파이프를 통해 받은 데이터를 변경시키고 그 결과를 파이프로 전송

13. 럼바우 객 동 기 + 동적모델링-상태도 , 기능모델링- 자료흐름도 : 동상기자

14. UML 다이어그램

- Activity diagram : 업무의 흐름을 모델링하거나 객체의 생명주기를 표현

-Sequence : 객체간의 메세지 전달을 시간적 흐름에서 분석

-Deployment: **기업 환경의** 구성과 컴포넌트들간의 관계

-Collaboration: **객체와 객체가** 주고받는 메세지 중심의 작성 동적 다이어그램

-Component: 소프트웨어 구조

-Class : 시스템의 구조적인 모습을 그림

15. 의존: 한개가 다른거에 영향, 실체화**:한 객체가 다른 객체에 의해 오퍼레이션 수행**,

일반화: 일반화된 사물과 좀 더 **특수화된 사물** 사이의 관계, (구체화\_

연관: 두사물간의 구조적 관계로 연결관계

16. CASE: 시스템 개발 과정의 일부 또는 전체를 자동화 시킨것

상위 CASE: **요구 분석과** 설계 단계를 지원 모순 검사, 모델 오류 검증, 자료 흐름도 작성

하위 CASE: **코드를 작성하고 테스트하며** 문서화하는 과정 지원

통합 CASE: 소프트웨어 개발 **주기 전체를 지원**

18. 에자일 개발 방법론

익스트림 프로그래밍, 스크럼크리스털 패밀리, 기능주도개발, **적응형소프트웨어 개발, 익스트림모델링**

19.GoF

생선패턴: 객체를 생성하는 것에 대한 패턴

구조패턴: 구조를 통해 확장성을 꾀하는 패턴

행위패턴: 행위의 변경,수정 등을 위한 패턴

singleton: 한 클래스에 한 객체만-생성

adaptor: 인터페이스가 호환되지 않는 클래스를 함께 이용가능하게 하는것 – 구조

decorator: 0개,1개 혹은 그 이상의 객체를 묶어 하나의 객체- 구조

state: 동일한 동작을 객체의 상태에 따라 다르게 처리- 행위

21. 힙정렬 O(nlog2n)

23. 살충제 패러독스: 동일한 테스트 케이스로 반복실행하면 새로운 결함 발견 못함

오류 부재의 궤변: 사용자의 요구사항을 만족못한다면 오류가 없어도 별로임

24.형상 감사: 기준선의 무결성을 평가하기 위해 확인, 검증, 검열 과정

체크아웃: 프로그램을 수정하기 위해 저장소에서 파일를 받아옴

체크인: 체크아웃한 파일의 수정을 완료한후 저장소의 파일을 새로운 버전으로 갱신

커밋: 체크인을 수행할때 이전에 갱신한 내용이 있으면 충돌을 알림

동기화: 일치시키는것

롤백: 데이터베이스에 오류가 발생할때 되돌리는것

25. 화이트 박스 테스트: **내부 구조와** **동**작을 검사하는 테스트 방식, 소프트웨어의 내부소스를 테스트, 내부의 논리적인 모든 경로를 테스트해 설계함(**화내지말라 동생**)

**블랙박스 테스트: 내부 구조나 작동원리를 모르는 상태에서 입력에 따른 출력테스트**

26. 단위 테스트 종류

명세기반 테스트: 주어진 명세를 빠짐없이 테스트 케이스로 구현하고 있는지 확인

구조기반 테스트: 프로그램 내부구조 및 복잡도 검증하는 화이트박스 테스트

성능 테스트: 애플리케이션의 처리량 응답시간, 경과시간, 성능 목표 달성 여부

27. 형상관리

- 소프트웨어에 가해지는 변경을 제어하고 관리

- 프로젝트계획, 분석서, 설계서 등 모두 관리대상

- 대표적인 GIThub 가있고 Gradle은 빌드 자동화 도구

- 유지 보수 단계말고도 개발 단계에서 적용가능

28. 디지털 저작권 관리(DRM)

- 콘텐츠 분배자: 암호화된 콘텐츠를 유통하는곳

- 패키저: 콘텐츠를 메타데이터와 함께 배포가능한 형태로 묶어 암호화

- 클리어링하우스: 키관리 및 라이센스 발급 관리

- DRM 컨트롤러: 배포된 콘텐츠의 이용권한을 통제

31. 버블 정렬 하나씩 다음꺼랑 비교

32. 정렬된 N개의 데이터 처리

O(NlogN) 은 퀵정렬, 병합정령

O(N^2)는 버블정렬, 삽입정렬 선택정렬

33. prefix: 전위 는 중간부터 infix: 중위 앞부터 postfix: 후위 마지막부터 🡪 인터넷 검색한번

텍스트이(가) 표시된 사진

자동 생성된 설명

34. EAI(Enterprise Application Integration) 기업내 상호 연동이 가능하게 해주는 솔루션

Point-to-Point 어플리케이션을 1:1연결 변경 및 재사용어려움

Hub&Spoke 단일 접점인 허브를 통해 데이터를 전송하는 중앙집중형 방식

Message Bus 애플리케이션 사이에 미들웨어를 두어 처리하는 방식

Hybrid 데이터 병목현상 최소화 두가지다 사용

35. Foxbase는 프로그래밍 언어

인터페이스 구현 검증도구: xUnit, STAF,water

39. 테스트는 오류를 찾는것 디버깅은 수정하는법

40. 선택정렬은 큐를 이용 재귀, 후위, 깊이는 스택

41. 수평 분할 기법이 아닌건 예측 분할

42. 시스템 카탈로그는 DBMS가 스스로 생성하고 유지함, INSERT, DELETE, UPDATE문 시스템 카탈로그 갱신 하는 것은 허용안됨

시스템 자신이 필요로 하는 스키마 및 여러가지 객체에 관한 정보를 포함

시스템 카탈로그에 저장되는 내용이 메타데이터

43. SELECT 뒤에 DISTINCT 를 적는다.

44. intersect : 교집합

49. 병행제어는 동시에 여러개의 트랜젝션을 병행 수행 할때 DB 일관성 파괴하지 않도록하는것

- 로킹기법, 타임스탬프 , 최적 병행 수행, 다중버전

시분할기법은 운영체제기법임.

51. 관계대수중 순수 관계 연산자 select, project, join, division

차집합은 일반 집합 연산자

53. 카티션 프로덕트는 차수는 둘이 **더하고** 카디널리티는 둘이 **곱해야한다.**

**카디널리티는 튜플의** 수를 나타낸다. **차수가 속성(애트리뷰트**)의 수

54. 검색이상: 종속성을 고려안하고 데이터 처리 연산 수행시 각종 이상 현상이 발생

갱신이상 : 반복된 데이터 중에 일부를 갱신하면 데이터의 불일치 발생

삽입이상: 불필요한 정보를 함께 저장하지 않고서는 정보 저장 불가능

삭제 이상: 필요한 정보를 함께 삭제하지 않고서는 어떤 정보를 삭제 불가능

60. 도부이결다조 정규형외우는법

61. Collision Domain: 충돌 발생을 검출할 수 있는 브리지 간 혹은 다른 계층 장치간의 이더넷 세그먼트 범위

CSMA/CA: 무선 랜에서 데이터 전송 시 매체가 비어 있음을 확인한 후 충돌을 회피하기 위해 임의 시간을 기다린 후 데이터를 전송하는 방식(AWAIT)

CSMA/CD: 충돌 감지 이더넷에서 각 단말이 전송 공유 매체에 규칙있게 접근하기 위한 매체 엑세스 제어방식(DETECTION)

**62. 결합도(Coupling) 강한것 부터**

**내 게 공 부 하라고 강요하지 말아요**

**제 가 스 트레스 받 자 나요**

**내용> 공동> 외부> 제어> 스탬프> 자료**

65. Detection: 교착상태 발생을 허용하고 발생 시 원인을 규명하여 해결(자원할당 그래프)

Avoidance: 교착상태 가능성을 배제하지 않고 적절하게 피해가기(은행원)

Recovery: 교착상태 발견 후 현황 대기를 배제 시키거나 자원을 중단하는 방법(선점, 프로세스중지)

Prevention: 교착상태의 필요조건을 부정함으로써 미리 예방(환형대기, 비선점, 점유와 대기, 상호배제)

66. CIDR 표기로 서브넷 마스크 구하기 🡪 ip주소/서브넷마스크

서브넷 마스크만큼 앞에서 1쓰고 남는 부분을 0넣고 8개씩 끊어서 계산하면됨

68. 반복, 스택, 부프로그램은 시간지역성과 연관

스래싱 방지는 각프로세스가 필요하는 프레임 제공

스래싱은 페이지 수행시간보다 교환시간이 커질때 발생

71. 리눅스 Bash쉘에서 export는 전역 변수처럼 사용하게되며 매개변수 없이 사용되면 환경변수 출력

변수를 출력하고자하면 echo$변수명

72. A는 아스키로 65 a는 아스키로 97

74.페이지 크기가 작으면 더 많은 페이징 사상테이블이 필요

내부 단편화 감소, 페이지 집합을 효율적으로 운영가능, 기억 장치의 효율이 높음, 총 입출력 시간 증가

75.응집도: 모듈 내부에서 기능적인 응집정도

우연적<논리적<시간적<절차적<교환적<순차적<기능적

76. TCP 프로토콜: 흐름제어, 양방향 가상회선, 응답데이터 전송가능

데이터링크: 인접한 노드사이의 프레임 전송 및 오류를 제어

77. c언어 연산자 우선순위

증감(++ --) >산술(\*/%)>산술(+-)>시프트(>> <<) > 관계연산자(>= =< < >) > 관계연산자( == !=) > 비트연산자(& ^ |) > 논리연산자(&& ||) > 조건연산자(? :) > 대입연산자 ( = += \*= /= %= <<= >>=)> 순서연산자(,)

78. c언어

stdlib.h 는 문자열을 수치 데이터로 바꾸는 문자 변환함수 수치를 문자열 바꿔주는 변환함수

strlen(): string.h 에 포함

표준 입출력은 stdio.h

수학관련식은 math.h

80. 데이터 링크 계층: 한 노드에서 다른 노드로 프레임을 전송하는 책임을 갖는 층

네트워크 계층: 패킷을 발신지로부터 최종 목적지까지 전달하는 책임, 패킷에 발신지와 목적지의 논리 주소 추가, 교환기능 제공

81. SAN: 네트워크상에 광채널 스위치의 이점인 고속 전송과 장거리 연결 및 멀티 프로토콜 기능을 활용, 동일 저장 장치 데이터 공유

MBR: 기억 장치를 출입하는 데이터가 잠시 기억되는 레지스터

NAC: MAC주소를 시스템에 등록, 일관된 보안관리 기능을 제공

NIC: 물리 계층과 데이터 계층의 서비스를 제공

82. SSH의 기본포트는 22번을 사용하며

전송되는 데이터는 암호화

키를 통한 인증은 공개키를 서버에 등록해야하며 서로 연결되어 있는 컴퓨터 간 원격 명령 실행이나 셀 서비스 수행

84. BLP 기밀성 모델: 불법적인 비밀 유출 방지에 중점 -기밀성강조

Clark-Wilson 무결성 모델: 자료의 변조 방지에 중점 – 무결성

Chinese Wall: 충돌을 야기시키는건 없어야한다. 이익의 충돌 금지

85. Sqoop: 하둡에서 데이터를 전송할 수 있도록 설계된 도구

86. 거리벡터 라우팅은 RIP

OSPF는 링크 상태 알고리즘 사용 , 네트워크 변화에 대응 멀티캐스팅 지원

87. COCOMO: 시스템의 비용을 산정하기 위해 시스템을 구성하고 있는 모듈과 서브시스템 비용 합산

Putnam: 소프트웨어 개발 주기의 간 단계별로 요구할 인력의 분포를 가정

Function-Point: 요구 기능을 증가 시키는 인자별로 부여하여 기능 점수 계산

PERT: 계획 평가 및 재검토 기술 일을 묘사하는데 사용

88. omission: 생략오류 한자리를 빼고기록

Transcription error: 필사오류 한자리를 잘못기록

Trasposition : 전위오류 좌우를 바꾸어기록

Double: 전위 오류두개

Additional: 한자리 추가

Random: 오류 두개

89. ISO 12207표준의 기본 생명주기: **공급 개발 운영 획득 유지보수 // 회사운영느낌**

90. COCOMO: Organic 5만 라인, Semi-Detached 30만프로젝트, Embedded // 30만 하ㅣ드웨어

91.SPICE 0~5

0단계 불안정

1단계 수행

2 관리

3 확립

4 예측

5 최적화

92.Meristor: 전원 공급이 끊어져도 전원이 공급되면 이전 상태 복원

MEMS: 다양한 기술로 미세 가공 후 전기 기계적 동작 가능케 한 초미세 장치

SNMP: 하드웨어 신기술 아님, 간이 망 관리 프로토콜

N-Screen: N개의 서로 다른 단말기에서 동일 콘텐츠 자유롭게 이용가능

93. 해쉬: 단방향 , SHA, MD5등

개인키: 양방향, stream(LFSR RC4), Block(DES, SEED) , 암호화 복호화 속도가 빠르며 알고리즘이 단순 파일크기 작음

공개키: 양방향, RSA, 키의 분배가 용이하고 관리해야할 키 수가 적음 대신 알고리즘 복잡 파일크기 큼

95.ASLR: 메모리를 적재할때 주소를 랜덤화 시켜서 공격자로 하여금 메모리 상의 주소를 추측하기 어렵게함

96. 스크린 서브넷: 외부 네트워크와 내부 네트워크 사이에 두는 완충적인 통신망

97.Secure OS

1. 식별 및 인증

2. 강제적 접근통제

3. 임의적 접근통제

4. 객체 재사용 방지

5. 완전한 중재 및 조정

6. 감사 및 감사 기록 축소

7. 안전한 경로

8. 보안 커널 변경 방지

9. 해킹 방지

10. 통합 관리

98. nmap: 서버에 열린 포트 정보를 스캐닝하여 보안 취약점을 찾는데 사용하는 도구

100. RSA 는 **공개키** 암호화 기법/ 공개키를 복호화 키는 **비밀키를** 사용

AES: 개인키 암호화기법/ 암호화 키와 **복호화 키가 동일**

**DSA 미국의 전자서명 표준**

ECC: 공개키 암호화 기법