

(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。 Int. Cl. (11) 공개번호 10-2006-0043629
G06Q 99/00 (2006.01) (43) 공개일자 2006년05월15일

(21) 출원번호 10-2005-0021265
(22) 출원일자 2005년03월15일

(30) 우선권주장 10/801,198 2004년03월15일 미국(US)

(71) 출원인 마이크로소프트 코포레이션
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원 마이크로소프트 웨이

(72) 발명자 테니스 데이비드 마이클
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내
새대트 캄란
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내
톨그 칼 이.에이.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내
포니 마이클 켈트
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내
고터머칼라 라마칸타채리 에스.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내
레씨 빅터 에스.
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내
리우 홍밍
미국 98052 워싱턴주 레드몬드 원 마이크로소프트 웨이
마이크로소프트 코포레이션 내

(74) 대리인 주성민
백만기
이중희

심사청구 : 없음

(54) 프로젝트 시간 및 경비 정보 획득 및 프로세싱 시스템

요약

네트워크를 거쳐 시간 및 경비 정보를 획득하고 그 정보를 프로젝트 회계 시스템 내로 프로세싱하기 위한 시스템은 프로젝트에 관련된 시간 및 경비 데이터를 저장하도록 적응된 프로젝트 회계 시스템을 갖는다. 사용자 인터페이스는 네트워크를 거쳐 브라우저를 통해 사용자에게 의해 액세스될 수 있는 서식을 제공한다. 서식은 사용자에게 의한 데이터 입력을 위한 필드들과 서식의 전자 제출을 위한 버튼을 포함한다. 서비스 API는 제출된 서식에 포함된 데이터에 기초하여 프로젝트 회계 시스템과의 트랜잭션을 인보크하도록 적응된다.

대표도

도 2

색인어

시간, 경비, 비용, 프로젝트, 회계, 객체

명세서

도면의 간단한 설명

도 1은 본 발명의 실시예들이 구현될 수 있는 컴퓨팅 시스템 환경의 개략도.

도 2는 본 발명의 실시예에 따른 시간 및 경비 시스템의 단순화된 블록도.

도 3은 본 발명의 실시예에 따라 보안, 데이터 액세스, 및 워크플로우 프로세스를 제공하는 비즈니스 어플리케이션 프레임워크에 대한 단순화된 블록도.

도 4a는 본 발명의 실시예에 따른 단순화된 타임시트 프로세싱 흐름도.

도 4b는 본 발명의 실시예에 따른 경비 보고 프로세싱 흐름도.

도 5는 본 발명의 실시예에 따른 통합된 모델링 언어 다이어그램 표기법에서의 엔터티 모델에 대한 단순화된 도면.

<도면의 주요부분에 대한 부호의 설명>

202: 서식

204: 웹 서버

206: 서비스 API

208: 데이터베이스

210: 네트워크

212: 사용자 인터페이스

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 일반적으로 사용자 인터페이스와 오피스 시스템 사이의 브리지를 제공하도록 적응된 시스템에 관한 것이다. 보다 상세하게, 본 발명은 프로젝트 관련 시간 및 경비 데이터를 프로젝트 회계 시스템 내로 획득하도록 적응된 시스템에 관련된 것이다.

발명이 이루고자 하는 기술적 과제

1980년대와 1990년대의 비즈니스 어플리케이션들은 주로 내부적인 후선업무(back-office)의 기능에 대한 비즈니스 프로세싱 자동화에 초점을 맞추었다. 내부적인 효율성이 중요하긴 하지만, 현 동향은 고객, 공급자, 파트너, 재무가 및 피고용인인 개인 및 비즈니스에 대한 비즈니스 내외적인 관계의 유효성의 최적화에 초점을 둔다. 전형적으로, 관계의 유효성을 최적화하면, 셀프서비스가 증대되고 서비스 제공이 보다 더 표적화된다. 여기에서, "셀프서비스"라는 용어는 다른 사람으로부터의 상호작용에 대한 필요없이, 사용자에게 기능 혹은 원하는 정보를 제공하도록 적응된 양방향의 사용자 인터페이스와 프리젠테이션 인터페이스를 일컫는다.

많은 비즈니스는 회계 소프트웨어 어플리케이션, 시간 기록 소프트웨어 어플리케이션 등을 포함한 그들의 비즈니스 운영을 지원하기 위하여 온-사이트 어플리케이션(on-site application)을 사용한다. 그러한 어플리케이션들은 종종 통합되지 않는다. 게다가, 셀프서비스를 제공하기 위하여, 웹 기반 서비스들이 종종 요구된다. 그러한 웹 기반 서비스들은 관련없는 비즈니스 어플리케이션들을 상호연결하기 위하여 종종 사용된다. 불행하게도, 그러한 상호연결은 종종 애드혹이다. 즉, 하나 이상의 웹 서비스를 사용하여 혹은 하나 이상의 웹이나 기존 소프트웨어 어플리케이션을 사용하여 임시변통으로 만들어질 수 있는 것이다.

회사가 효율적으로 고객들에게 청구하고, 비용을 이해하고, 피고용인에게 변상하기 위하여, 회사는 모든 프로젝트 관련 비용과 시간 사항을 획득해야 한다. 전형적으로, 피고용인들은 자신의 시간 및 경비를 기록하고, 그 시간 및 경비 데이터는 회계 시스템으로 입력된다. 그러한 시스템은 데이터를 정확하게 수신하고 기록하는 것이 중요하다.

다양한 회사들이 시간 및 경비 기능성을 프로젝트 회계 시스템과 통합한다. 그러나 그러한 시스템들은 종종 많은 조직에 대해 너무 비용이 많이 드는 기업 솔루션일 수 있다.

발명의 구성 및 작용

네트워크를 거쳐 비용 정보를 획득하고 그 정보를 프로젝트 회계 시스템 내로 프로세싱하기 위한 시스템은 회계 시스템, 사용자 인터페이스 및 서비스 API를 포함한다. 프로젝트 회계 시스템은 프로젝트와 관련된 시간 및 경비 데이터를 저장하기에 적합하도록 적응된다. 사용자 인터페이스는 브라우저를 통해 사용자에게 의해 네트워크를 거쳐 액세스하기 쉬운 서식을 제공한다. 서식은 사용자에게 의한 데이터 입력을 위한 필드와 서식의 전자 제출을 위한 버튼을 포함한다. 서비스 API는 제출된 서식에 포함된 데이터에 기초하여 프로젝트 회계 시스템과의 트랜잭션을 인보크하도록 적응된다.

또다른 실시예에서, 서식을 통해 회계 데이터베이스로 시간 및 경비 데이터를 획득하는 방법이 본 발명의 실시예에 따라 제공된다. 서식들은 웹 서버에서 호스팅된다. 각 서식은 브라우저를 통해 네트워크를 거쳐 사용자에게 액세스 가능하다. 각 서식은 사용자 데이터 입력을 위한 데이터 필드, 사용자에게 의해 완료되면 웹 서식을 제출하도록 적응된 객체 및 사용자에게 의한 제출 시에 비즈니스 룰을 인보크하도록 적응된 내장 서버 컨트롤을 포함한다. 비즈니스 룰은 관리된 코드로 작성된다. 제출된 서식은 인보크된 비즈니스 룰에 따라 서비스 API를 사용하여 프로세싱된다. 서비스 API는 서식 내에 포함된 데이터를 회계 데이터베이스 내의 엔터티들로 연결한다.

또다른 실시예에서, 네트워크를 거쳐 시간 및 경비 정보를 획득하고 정보를 회계 시스템으로 프로세싱하기 위한 시스템은 회계 시스템, 웹 파트 및 서비스 API를 포함한다. 회계 시스템은 시간 및 경비 정보를 저장하도록 적응된다. 대다수의 웹 파트 서식이 네트워크를 거쳐 브라우저를 통한 사용자 입력을 위해 적응된다. 서비스 API는 사용자 입력을 회계 시스템으로 프로세싱하기 위하여 관리된 코드로 작성된 비즈니스 룰을 구현하고 시퀀스화하도록 적응된다.

본 발명을 상세히 설명하기에 앞서, 본 발명이 사용될 수 있는 환경의 한 실시예가 기술된다. 도 1은 본 발명이 구현될 수 있는 적합한 컴퓨팅 시스템 환경(100)의 예를 도시한다. 컴퓨팅 시스템 환경(100)은 적합한 컴퓨팅 환경의 한 예일 뿐이며 본 발명의 사용이나 기능의 범위에 대해 어떠한 제한을 가하고자 하는 것이 아니다. 컴퓨팅 환경(100)은 예시적인 운영 환경(100)에 도시된 컴포넌트들 중 어떤 하나 혹은 조합에 관련된 어떠한 의존성이나 요구사항을 가진 것으로 해석되어서는 안 된다.

본 발명은 다양한 기타의 범용 혹은 특별한 목적을 갖는 컴퓨팅 시스템 환경이나 구조에서 동작 가능하다. 본 발명에서 사용하기에 적합할 수 있는 잘 알려진 컴퓨팅 시스템, 환경 및/또는 구조의 예는 개인용 컴퓨터, 서버 컴퓨터, 포켓용 또는 랩탑 장치, 멀티프로세서 시스템, 마이크로 프로세서 기반 시스템, 셋탑 박스, 프로그래밍 가능한 가전기기, 네트워크 PC, 미니컴퓨터, 메인프레임 컴퓨터, 전화통신 시스템, 상술된 시스템이나 장치 중 어떠한 것을 포함하는 분산 컴퓨팅 환경 등을 포함하지만 제한되는 것은 아니다.

본 발명은 프로그램 모듈과 같이 컴퓨터에 의해 실행될 수 있는 컴퓨터 실행가능한 명령어의 일반적인 문맥으로 기술될 수 있다. 일반적으로, 프로그램 모듈은 특정 태스크를 수행하거나 특정 추상 데이터 타입을 구현하는 루틴, 프로그램, 객체, 컴포넌트, 데이터 구조 등을 포함한다. 본 발명은 또한 통신 네트워크를 통해 연결된 원격 프로세싱 장치에 의해 태스크가 수행되는 분산 컴퓨팅 환경에서 실행될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 메모리 저장 장치를 포함하여 로컬 및 원격 컴퓨터 저장 매체 모두에 저장될 수 있다.

도 1을 참조하면, 본 발명을 구현하기 위한 예시적인 시스템은 컴퓨터(110)의 형태로 범용 컴퓨팅 장치를 포함한다. 컴퓨터(110)의 컴포넌트들은 프로세싱 유닛(120), 시스템 메모리(130) 및 시스템 메모리를 포함한 다양한 시스템 컴포넌트들을 프로세싱 유닛(120)으로 연결하는 시스템 버스(121)를 포함하지만 제한되지는 않는다. 시스템 버스(121)는 메모리 버스나 메모리 컨트롤러, 주변 버스 및 다양한 버스 구조 중 어떤 것을 사용하는 로컬 버스를 포함하는 여러 형태의 버스 구조 중 어떤 것일 수 있다. 제한을 가하지 않는 예로서, 그러한 구조는 산업 표준 구조(Industry Standard Architecture:ISA) 버스, 마이크로 채널 구조(Micro Channel Architecture:MCA) 버스, 강화된 ISA(EISA) 버스, 비디오 전자 표준 협회(Video Electronics Standards Association:VESA) 로컬 버스 및 Mezzanine 버스로도 알려진 주변 컴포넌트 인터커넥트(Peripheral Component Interconnect:PCI) 버스를 포함하며, 제한되지는 않는다.

컴퓨터(110)는 전형적으로 다양한 컴퓨터 판독가능 매체를 포함한다. 컴퓨터 판독가능 매체는 컴퓨터(110)에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리가능 및 분리불가능 매체 모두를 포함한다. 제한을 가하지 않는 예로서, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체와 통신 매체로 구성될 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 방법 및 기술로 구현될 수 있는 휘발성 및 비휘발성, 분리가능 및 분리불가능 매체 모두를 포함한다. 컴퓨터 저장 매체는 RAM, ROM, EEPROM, 플래시 메모리 또는 기타 메모리 기술, CDROM, DVD 또는 기타 광학 디스크 저장 장치, 자기 카세트, 자기 테이프, 자기 디스크 저장 장치 또는 기타 자기 저장 장치, 또는 원하는 정보를 저장하기 위해 사용될 수 있고 컴퓨터(110)에 의해 액세스될 수 있는 기타 매체를 포함하며, 제한되지는 않는다. 통신 매체는 전형적으로 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 반송파나 기타 전송 메커니즘과 같은 변조된 데이터 신호에서의 기타 데이터를 포함하며, 어떠한 정보 전달 매체를 포함한다. "변조된 데이터 신호"라는 용어는 신호에 있는 정보를 암호화하기 위한 방식으로 세팅 또는 변화시킨 하나 이상의 특성을 갖는 신호를 의미한다. 제한을 가하지 않는 예로서, 통신 매체는 유선 네트워크나 직접 유선연결된 연결과 같은 유선 매체와 청각, RF, 적외선 및 기타 무선 매체와 같은 무선 매체를 포함한다. 상기 어떠한 것의 조합 또한 컴퓨터 판독가능 매체의 범위에 포함되어야 한다.

시스템 메모리(130)는 ROM(131)과 RAM(132) 같은 휘발성 및/또는 비휘발성 메모리의 형태로 컴퓨터 저장 매체를 포함한다. 시작할 때와 같이 컴퓨터(110) 내의 구성요소들 사이에 정보 교환을 돕는 기본적인 루틴들을 포함하는 기본적인 입력/출력 시스템(133)(BIOS)은 전형적으로 ROM(131)에 저장된다. RAM(132)은 전형적으로 직접 액세스 가능하고/하거나 현재 프로세싱 유닛(120)에 의해 동작되고 있는 데이터 및/혹은 프로그램 모듈을 포함한다. 제한을 가하지 않는 예로서, 도 1은 운영 시스템(134), 어플리케이션 프로그램(135), 기타 프로그램 모듈(136) 및 프로그램 데이터(137)를 도시한다.

컴퓨터(110)는 또한 다른 분리가능/분리불가능, 휘발성/비휘발성 컴퓨터 저장 매체를 포함한다. 단지 예로서, 도 1은 분리불가능한 비휘발성 자기 매체를 판독하거나 기록하는 하드 디스크 드라이브(140), 분리가능한 비휘발성 자기 디스크(152)를 판독하거나 기록하는 자기 디스크 드라이브(151), 그리고 CD ROM이나 기타 광학 매체와 같은 분리가능한 비휘발성 광 디스크(156)를 판독하거나 기록하는 광 디스크 드라이브(155)를 도시한다. 예시적인 운영 환경에서 사용될 수 있는 기타 분리가능/분리불가능, 휘발성/비휘발성 컴퓨터 저장 매체는 자기 테이프 카세트, 플래시 메모리 카드, DVD, 디지털 비디오 테이프, 고체 상태 RAM, 고체 상태 ROM 등을 포함하지만 제한되지는 않는다. 하드 디스크 드라이브(141)는 전형적으로 인터페이스(140)와 같은 분리불가능 메모리 인터페이스를 통해 시스템 버스(121)로 연결되고, 자기 디스크 드라이브(151)와 광 디스크 드라이브(155)는 전형적으로 인터페이스(150)와 같은 분리가능 메모리 인터페이스에 의해 시스템 버스(121)로 연결된다.

상술되고 도 1에 도시된 드라이브들과 관련 컴퓨터 저장 매체는 컴퓨터(110)에 대해 컴퓨터 관독가능 명령어들, 데이터 구조, 프로그램 모듈, 기타 데이터의 저장을 제공한다. 예를 들어 도 1에서, 하드 디스크 드라이브(141)는 운영 시스템(144), 어플리케이션 프로그램(145), 기타 프로그램 모듈(146) 및 프로그램 데이터(147)를 저장하는 것으로 도시된다. 이러한 컴포넌트들은 운영 시스템(134), 어플리케이션 프로그램(135), 기타 프로그램 모듈(136) 및 프로그램 데이터(137)와 같을 수도 있고 다를 수도 있다는 점을 유의해야 한다. 운영 시스템(144), 어플리케이션 프로그램(145), 기타 프로그램 모듈(146) 및 프로그램 데이터(147)는 여기서 최소한 그것들이 다른 것들이라는 것을 도시하기 위하여 다른 번호를 부여받는다.

사용자는 키보드(162), 마이크(163) 및 마우스, 트랙볼 또는 터치 패드와 같은 포인팅 장치(161)와 같은 입력 장치를 통해 컴퓨터(110)로 명령어와 정보를 입력할 수 있다. 기타 입력 장치들(도시되지 않음)은 조이스틱, 게임 패드, 위성 디쉬, 스캐너 등을 포함할 수 있다. 이러한 장치들과 기타 입력 장치들은 종종 시스템 버스로 연결되는 사용자 입력 인터페이스(160)를 통해 프로세싱 유닛(120)으로 연결되는데, 기타 인터페이스 및 병렬 포트, 게임 포트 또는 USB와 같은 버스 구조에 의해 연결될 수 있다. 모니터(191)나 다른 종류의 디스플레이 장치 또한 비디오 인터페이스(190)와 같은 인터페이스를 통해 시스템 버스(121)로 연결된다. 모니터 외에, 컴퓨터는 스피커(197)와 프린터(196) 같은 다른 주변 출력 장치들을 포함할 수 있고, 그것은 출력 주변 인터페이스(195)를 통해 연결될 수 있다.

컴퓨터(110)는 원격 컴퓨터(180)와 같은 하나 이상의 원격 컴퓨터들로의 논리적 연결을 사용하는 네트워크 환경에서 동작할 수 있다. 원격 컴퓨터(180)는 개인용 컴퓨터, 포켓용 장치, 서버, 라우터, 네트워크 PC, 피어 디바이스나 기타 공통 네트워크 노드일 수 있으며, 전형적으로 컴퓨터(110)와 관련되어 상술된 구성요소들 중 다수 혹은 모두를 포함한다. 도 1에 도시된 논리적 연결들은 근거리 통신망(LAN)(171)과 광역 통신망(WAN)(173)을 포함하지만, 다른 네트워크들도 포함할 수 있다. 그러한 네트워킹 환경들은 사무실, 기업 규모 컴퓨터 네트워크, 인트라넷 및 인터넷에서 흔한 것이다.

LAN 네트워킹 환경에서 사용될 때, 컴퓨터(110)는 네트워크 인터페이스나 어댑터(170)를 통해 LAN(171)으로 연결된다. WAN 네트워킹 환경에서 사용될 때, 컴퓨터(110)는 전형적으로 인터넷과 같은 WAN(173)을 거쳐 통신을 설정하기 위한 모뎀(172)이나 다른 수단을 포함한다. 내장되거나 외장일 수 있는 모뎀(172)은 사용자 입력 인터페이스(160)나 다른 적합한 메커니즘을 통해 시스템 버스(121)로 연결될 수 있다. 네트워크 환경에서, 컴퓨터(110)와 관련되어 기술된 프로그램 모듈이나 그들의 일부는 원격 메모리 저장 장치에 저장될 수 있다. 제한을 가하지 않는 예로서, 도 1은 원격 어플리케이션 프로그램들(185)을 원격 컴퓨터(180)에 상주하는 것으로 도시한다. 도시된 네트워크 연결들은 예시이고, 컴퓨터들 사이에 통신 링크를 설정하기 위한 다른 수단들이 사용될 수 있다는 것을 잘 알 수 있을 것이다.

도 2는 본 발명의 실시예에 따른 시간 및 경비 시스템(200)을 도시한다. 웹 파트 서식(202)이 웹 서버(204) 내에 배치된다. 서비스 어플리케이션 프로그램 인터페이스(API)(206)는 관리된 코드(managed code)로 작성된 비즈니스 룰을 구현하고 시퀀스화한다. 필요에 따라, 서비스 API(206)는 데이터베이스(208)와의 트랜잭션 및/또는 쿼리를 인보크한다. 서식(202)은 네트워크(210)를 거쳐 사용자 인터페이스(212)에 디스플레이된다.

일반적으로, 서비스 API(206)는 소정의 다른(전형적으로 더 낮은 레벨의) 모듈에 의해 제공된 서비스를 액세스하기 위해 어플리케이션 프로그램이 사용하는 인터페이스(또는 규약을 호출하는 함수의 세트)이다. 이러한 예에서, 서비스 API(206)는 관리된 코드로 작성된 비즈니스 룰을 프로세싱하고 선택적으로 데이터베이스(208)에 의해 제공된 서비스들을 액세스하기 위하여 사용된다.

"관리된 코드"라는 용어는 런타임을 표적으로 하는 언어 컴파일러를 이용하여 개발된 코드를 일컫는다. 관리된 코드는, 런타임이 어셈블리 모듈에서 암호화된 방법을 찾아내고, 보안 정보를 저장 및 검색하며, 예외를 프로세싱하는 것 등을 허용하기 위하여 정보(때때로 "메타데이터"로 언급됨)를 제공한다. 일반적으로, 관리된 코드는 런타임을 표적으로 하는 코드이다. 런타임은 실행 시에 코드를 관리하여 메모리 관리, 스레드 관리 및 리모팅(모듈들이 시스템 경계를 가로질러 통신할 수 있게 해주는 API)과 같은 핵심 서비스들을 제공하면서, 또한 코드 내에서 엄격한 형태 안정성을 강요한다.

관리된 코드 실행 환경 내에서의 프로그래밍은 본 기술 분야에서 알려져 있다. 관리된 코드의 알려진 한 예는 공통 언어 런타임(Common Language Runtime)(CLR)이다. 컴파일러와 튜블 런타임의 기능성을 노출시키고, 개발자가 관리된 코드 실행으로부터 이득을 얻는 코드를 작성할 수 있게 한다. 관리된 코드는 언어 간 통합, 언어 간 예외 처리, 강화된 보안, 버전 관리 및 전개 지원과 같은 특성들로부터 이득을 얻는다. 또한, 관리된 코드는 서비스를 디버깅하고 프로파일링할 뿐 아니라, 컴포넌트 상호작용에 대한 단순화된 모델을 제공한다.

네트워크(210)는 데이터 전송을 지원할 수 있는 임의의 유형의 네트워크일 수 있으며, 전자 교환 전화 통신망, 근거리 통신망, 광역 통신망(인터넷을 포함함), 셀룰러 또는 디지털 무선 통신망, 혹은 기타 유형의 네트워크를 포함한다. 본 발명은 전송 매체가 데이터 전송을 지원하는 한, 전송 매체에는 독립적이다.

일반적으로, 사용자 인터페이스(212)는 서식(202)을 네트워크(210)를 거쳐 사용자에게 디스플레이하기 위한 프리젠테이션 인터페이스를 제공한다. 예시적인 실시예에서, 사용자 인터페이스(212)는 인터넷 브라우저 어플리케이션의 브라우저 윈도우 내의 페이지로 디스플레이된다. 사용자는 사용자 인터페이스(212) 내에서 서식(202)을 작성하고, 네트워크(210)를 거쳐 웹 서버(204)로 서식(202)을 제출한다. 서식(202)은 서비스 API(206)를 호출한다. 호출된 서비스 API(206)가 인보크되고, 서식(202) 중 하나를 통해 사용자에게 의해 제출된 데이터를 관리된 코드에 따라 프로세싱한다. 필요하다면, 서비스 API(206)는 예를 들어 피고용인으로부터의 시간 입력을 저장하기 위하여 데이터베이스(208)와 상호작용한다.

일반적으로, 서비스 API(206)는 관리된 코드로 작성된 비즈니스 룰을 프로세싱한다. 비즈니스 룰은 승인 프로세스에서의 단계들을 포함할 수 있다. 따라서, 비즈니스 룰은 워크플로우 프로세스를 정의할 수 있다. 일반적으로, 서비스 API(206)는 예를 들어 회계 서브시스템이 송장 생성, 지불 획득, 피고용인 임금 및 경비 변상을 완료하기 위해 시간 및 경비 트랜잭션을 수신하고 시간 및 경비 데이터를 프로세싱하기 위한 필수적인 중요한 링크를 완료한다.

일반적으로, 승인 프로세스는 다양한 방식으로 구현될 수 있다. 일 실시예에서, 승인 프로세스는 자동화된 워크플로우의 일부이다. 서비스 API(206)는 웹 서식에 저장된 데이터에 대하여 비즈니스 룰을 프로세싱한다. 관리자가 시스템에 로그인할 때, 관리자는 자신이 감독 기능을 수행하는 게시된 시간 및 경비 서식을 경고받을 수 있다. 그러면, 관리자는 그러한 서식들을 보고, 서식을 수락, 종료하거나 혹은 추가의 프로세싱을 위해 피고용인에게 리턴할 수 있다. 피고용인 사용자가 시스템에 로그인할 때, 피고용인은 변경을 요구하는 서식에 액세스하고, 계류중인 서식을 변경하며, 새로운 서식을 생성하고 제출하는 것 등을 할 수 있다. 시스템은 각 사용자의 시스템 내에서의 "인증된 역할"에 따라 각 사용자에게 관련된 서식으로의 액세스를 허용할 수 있다. 예를 들어, 한 피고용인은 한 조직 내에서 다수의 역할을 가질 수 있다. 한 피고용인이 어느 피고용인에 대해 감독관일 수 있고, 또 다른 피고용인에 대해서는 관리자 동일 수 있다.

도 3은 컴포넌트 구성요소들에 기반을 둔 비즈니스 어플리케이션 프레임워크(300)를 도시한다. 일반적으로, 비즈니스 어플리케이션 프레임워크(300)는 데이터를 프로세싱하고 저장하며 디스플레이하기 위한 일련의 어플리케이션 서비스들을 제공한다. 비즈니스 어플리케이션 프레임워크(300)는 보안, 데이터 액세스, 메세징, 워크플로우, 이벤트 중개 등에 대한 지원을 제공한다.

비즈니스 어플리케이션 프레임워크(300)는 엔터티(304) 및 프로세스(306)와 상호작용하는 사용자 인터페이스(UI)(302)를 포함한다. UI(302)는 데이터를 입력하고, 태스크를 수행하며, 정보를 디스플레이/보고하기 위한 인터페이스를 제공한다. 엔터티(304)는 비즈니스 데이터와 데이터 무결성 룰 둘 다를 포함하는 데이터의 기본 유닛이다. 프로세스(306)는 비즈니스 로직의 기본 유닛이다.

정책(308)은 보안, 데이터 액세스, 워크플로우 및 이벤트 중개를 제공하기 위한 일련의 비즈니스 룰 맞춤 포인트(business rule customization point)이다. 정책(308)은 UI(302)로 디스플레이된 데이터를 관리하기 위하여 엔터티(304) 및 프로세스(306)와 직접 상호작용한다. 후선업무 어플리케이션은 프레임워크(300)의 많은 동작에 대한 초점이다.

일반적으로, 비즈니스 어플리케이션 프레임워크(300)는 컴포넌트들에 기반을 둔다. 각 컴포넌트는 0개 이상의 엔터티들 및 0개 이상의 프로세스들로 구성된다. 컴포넌트들은 일련의 클래스들에 대한 인터페이스를 제공한다. 확장성 생성 언어(Extensible Markup Language)(XML) 웹 서비스들이 컴포넌트의 공개 인터페이스의 모두 혹은 일부를 사용함으로써 생성된다.

도 4a 및 4b는 각각 타임시트와 경비 보고 프로세싱 흐름도이다. 본 실시예에서, 두 흐름도는 UI에 표시된 옵션을 통해 사용자에게 의해 액세스될 수 있고, 두 흐름도 모두 데이터를 비용청구 함수(billing function)로 보낸다. 따라서, 두 흐름도는 공통의 구성요소들과 관련되고 그것들을 공유한다. 대안적인 실시예에서, 두 흐름도들은 분리될 수 있고, 프로세싱된 타임시트와 경비 보고 데이터를 일반 원장(general ledger)으로 보내기 위해, 부가적인 워크플로우 프로세스가 이용될 수 있다.

도 4a는 본 발명의 실시예에 따른 타임시트 프로세싱 흐름에 대한 단순화된 흐름도로서, 사용자는 웹 브라우저를 사용하여 인터페이스에 액세스하고, 인터페이스는 웹 기반 형태로 디스플레이된다. 사용자 인터페이스(402)는 사용자에게 세 가지 기본 타임시트 옵션 - 타임시트 생성(404), 타임시트 삭제(406) 및 기존 타임시트 변경(408) - 을 제공한다.

사용자가 UI(402)를 통해 타임시트를 생성(404)하기 위한 옵션을 선택하면, 타임시트 서식이 브라우저 윈도우 내에 디스플레이된다. 사용자는 타임시트 서식을 완성하고, 완성된 타임시트(410)를 제출한다. 서비스 API는 비즈니스 룰을 프로세싱하고, 그것은 승인 프로세스(412)를 시작한다. 앞서 논의된 것처럼, 비즈니스 룰은 워크플로우 프로세스를 포함할 수 있고, 워크플로우 프로세스는 승인을 위해 감독관에게 타임시트를 제시하는 것과 같은 인증 단계를 포함한다. 그 후 감독관은 타임시트를 승인(414), 종료(416)하거나, 변경을 위해 사용자에게 리턴(418)한다. 만약 타임시트가 사용자에게 리턴(418)되면, 사용자는 기존 타임시트를 변경(408)하고 변경된 타임시트를 제출(410)할 수 있다.

만약 감독관이 타임시트를 승인(414)한다면, 타임시트 데이터는 프로젝트 회계 시스템으로 전송(420)되고, 임금 시스템으로 전송(422)된다. 회계 시스템 및 임금 시스템으로부터의 타임시트 데이터는 그 후 일반 원장으로 전달(424)되고, 비용 청구 함수로 전달(426)된다.

만약 UI(402)로부터의 사용자 혹은 승인 프로세스(412)로부터의 감독관이 타임시트를 종료(416)하기로 결정한다면, 타임시트 데이터는 삭제된다.

비용 청구 함수(426)는 비용 청구 출력정보나 전자 계산서를 직접 생성할 수 있고, 각 고객에 대한 비용 청구 정보를 후선 업무 어플리케이션으로 전달할 수 있다.

도 4b는 본 발명의 실시예에 따른 경비 보고 프로세싱 흐름을 도시한다. UI(402)로부터 사용자가 세 가지 옵션 - 경비 보고 생성(432), 경비 보고 삭제(434) 또는 경비 보고 변경(436) - 중에서 선택한다.

UI(402)를 통해 사용자가 경비 보고를 생성(432)하기 위한 옵션을 선택하면, 경비 보고 서식이 브라우저 윈도우 내에 디스플레이된다. 사용자는 경비 보고 서식을 완성하고, 완성된 경비 보고를 제출(438)한다. 서비스 API는 비즈니스 룰을 프로세싱하고, 그것은 승인 프로세스를 시작(440)한다(도 4a의 비즈니스 룰(412)과는 별개의 비즈니스 룰일 수 있다). 앞서 논의된 것과 같이, 비즈니스 룰은 승인을 위해 감독관에게 경비 보고를 제시하는 것과 같은 인증 단계를 포함하는 워크플로우 프로세스를 포함할 수 있다. 감독관은 그 후 경비 보고를 승인(442), 종료(444)하거나, 변경(436)을 위해 사용자에게 리턴(446)할 수 있다. 만약 경비 보고가 사용자에게 리턴(446)된다면 사용자는 경비 보고를 변경(436)할 수 있고 변경된 경비 보고를 제출(438)할 수 있다.

만약 감독관이 경비 보고를 승인(442)한다면, 경비 보고 데이터는 프로젝트 회계 시스템으로 전송(448)된다. 회계 시스템으로부터의 경비 보고 데이터는 그 후 일반 원장으로 전달(450)되고, 비용 청구 함수(426)로 전달된다.

만약 UI(402)로부터의 사용자 혹은 승인 프로세스(440) 동안 감독관이 경비 보고를 종료(444)하기로 결정한다면 경비 보고 데이터는 삭제된다.

비용 청구 함수(426)는 비용 청구 출력정보나 전자 계산서 서식을 직접 생성할 수 있고, 또는 각 고객에 대한 비용 청구 정보를 후선 업무 어플리케이션으로 전달할 수 있다.

일반적으로, 사용자는 웹 서버를 통해 타임시트 및 경비 보고 서식에 액세스할 수 있다. 사용자는 웹 브라우저 윈도우 내에서 서식을 완성하고, 서식을 전송한다. 서식은 서비스 API를 호출하는데 그것은 서식 내에 포함된 데이터를 프로세싱하기 위하여 관리된 코드로 작성된 비즈니스 룰을 구현하고 시퀀스화한다. 최종적으로, 데이터는 정확한 재정 데이터를 생성하고 유지하기 위하여 저장 및/또는 이용된다.

본 발명에 의해 제공된 시간 및 경비 특성은 회사가 자신의 고객들에게 효율적으로, 적시에 그리고 적절한 방식으로 비용을 청구할 수 있도록 모든 프로젝트 관련 비용을 획득한다. 또한, 회사는 프로젝트로 인해 발생된 경비에 대해 피고용인들에게 변상하기 위하여 정확하고 효율적으로 경비 데이터를 획득할 수 있다. 시간은 프로젝트 및 비프로젝트 모두의 목적으로 획득되고, 피고용인들이 지불받을 수 있도록 임금 시스템으로 전송된다. 본 발명에 기술된 시간 및 경비 기능은 많은 웹 서버와 구현들과 함께 이용될 수 있는 셀프서비스 웹 어플리케이션을 제공한다. 일 실시예에서, 서식들은 ASP.Net 웹 파트 서식일 수 있고, 그것은 ASP.NET 서버 컨트롤을 포함한다. 웹 파트 서식은 마이크로소프트® 웨어포인트® 제품 및 기술과 관련된 페이지일 수 있다. "마이크로소프트" 및 "웨어포인트"라는 용어는 워싱턴주 레드몬드 시의 마이크로소프트 코포레이션에 의한 고유 상표로 등록된다. 설계 시에 프로그래머에 의해 웹 서식 페이지로 추가되는 표준 ASP.NET 컨트롤과는 달리, 웹 파트는 실행 시에 사용자에게 의해 웹 파트 페이지에 있는 웹 파트 존(Web Part Zone)으로 추가되도록 의도된다. 사용자에게 할당된 권한 허가에 따라, 사용자는 매우 자유롭게 웹 파트 및 웹 파트 페이지를 변경할 수 있다. 다른 실시예에서, 서식들은 또한 다른 형태의 서식일 수 있다.

본 발명은 예를 들어 마이크로소프트® 비즈니스 솔루션 비즈니스 포털(Microsoft® Business Solutions Business Portal)을 이용하여 ASP.NET 및 그레이트 플레인 어카운팅 백-엔드 시스템(Great Plains Accounting back-end system)으로 구현될 수 있다. 다른 실시예에서, ASP 서버는 비즈니스 룰을 제공하고 프로세싱하며 데이터베이스와 상호 작용하기 위하여 ASP 서식 및 관련 서비스 API를 제공받을 수 있다.

일반적으로, 본 발명은 단 한번 입력된 시간 및 경비 데이터를 이용하여 발생된 프로젝트 비용을 획득하고, 피고용인에게 변상하며, 고객에 대한 송장을 작성하는 엔드 투 엔드 비즈니스 프로세스(end-to-end business process)에서 중요한 링크를 제공한다. 본 발명의 중요한 이점은 고객에 대한 송장을 작성하는 데에 걸리는 시간을 감소시키고 따라서 지불을 수신하는 데에 걸리는 시간을 감소시키는 능력이다. 이것은 회사의 현금의 흐름을 개선하는 잠재력을 갖는다. 또한, 피고용인은 비자동화된 혹은 통합되지 않은 시간 및 경비 획득 시스템이나 어플리케이션과 비교하여 더욱 신속하게 현금 경비에 대해 변상받을 수 있다.

승인 프로세스가 도 4a 및 4b에서는 감독관을 개입시키는 것으로 도시되었지만 본 발명의 기술분야에서 숙련된 지식을 가진 자는 승인 프로세스가 소정의 지출 및/또는 시간 기록에 대해 자동화될 수 있다는 점을 이해해야 한다. 따라서, 소정의 경우에, 데이터는 관리된 코드로 구현된 비즈니스 룰에 따라, 감독관의 개입없이 프로그램적으로 프로세싱되고 승인될 수 있다.

도 5는 본 발명의 실시예에 따라 통합 모델링 언어(UML)를 사용하는 엔터티 모델을 도시한다. 일반적으로, 도면 내에서 속이 빈 끝을 가지는 화살표는 베이스 클래스로의 관계를 나타내고, 반면에 얇은 화살표는 베이스 클래스 이외의 관계를 나타낸다. 얇은 화살표에 인접한 숫자는 객체가 반드시 포함되어야 하는지 또는 선택사항인지를 나타낸다. 화살표 다음에 표시된 "1"은 객체가 반드시 포함되어야 한다는 것을 나타내고, 반면 "0..1"의 표시는 0 또는 1개의 객체가 포함될 수 있다는 것을 의미한다. "0..n"은 이론상 0 또는 그 이상의 객체들이 포함될 수 있다는 것을 나타낸다.

도시된 바와 같이, UML 비즈니스 엔터티 모델(500)은 경비 헤더 베이스(502)와 타임시트 헤더 베이스(504)를 갖는다. 도시된 것처럼, 경비 라인(506)은 경비 라인 베이스 클래스(508)로부터 구조를 상속받는다. 각 경비 라인(506)은 경비 헤더(510)를 가져야 하고, 그것은 경비 헤더 베이스(502)로부터 구조를 상속받는다. 차례로, 경비 헤더(510)는 적어도 하나의 피고용인 객체(512)를 포함해야 한다. 피고용인 객체(512)는 회사의 피고용인을 나타낸다. 헤더는 선택적으로 피고용인의 관리자 및/또는 감독관을 포함할 수 있으며, 그들 또한 피고용인 객체이다.

경비 라인 객체(506)는 선택적으로 프로젝트 객체(514)와 비용 카테고리(516)를 포함할 수 있다. 따라서, UML은 시스템의 구조에 대한 객체 기반 표현을 정의한다. 각 구성요소는, 클래스의 인스턴스화가 객체가 그 성질을 상속받는 객체에 대한 베이스 구성자를 요구하는 시스템 내의 소프트웨어 클래스로 여겨질 수 있다.

비즈니스 경비와 같이, 타임시트 라인(518)은 타임시트 라인 베이스(520)로부터 구조를 상속받는다. 타임시트 라인(518)은 타임시트 헤더(522)를 포함할 수 있고, 그것은 타임시트 헤더 베이스(504)로부터 구조를 상속받는다. 경비 헤더(510)와 같이, 타임시트 헤더(522)는 피고용인 객체(512)를 포함해야 하고, 피고용인 객체(512)에 관련된 관리자와 감독관을 포함할 수 있다.

타임시트 라인(518)은 선택적으로 프로젝트 객체(514)와 비용 카테고리 객체(516)를 포함할 수 있다. 따라서, 타임시트 라인 객체(518)는 회계 데이터베이스에 포함된 타임시트 데이터에 관련된 구성요소들로 구성된다. 특정 구현에 따라, 기타 객체들 및 관계들이 특정 구성에 맞추어 정의될 수 있다.

일반적으로, 서비스 API는 데이터베이스로 및 데이터베이스로부터 데이터 트랜잭션을 인보크한다. 서비스 API는 쿼리 트랜잭션을 인보크할 수 있고, 비즈니스 엔터티를 데이터베이스로 저장하는 등을 할 수 있다. 비즈니스 엔터티들은 객체 지향 접근 방식을 사용하여 데이터베이스에 저장된 정보에 액세스하기 위하여 사용된 추상적 개념이다.

본 발명이 특정 실시예를 참조하여 기술되고 있지만, 본 발명의 기술분야에서 숙련된 지식을 가진 자는 본 발명의 취지와 범위를 벗어나지 않고 형태와 세부항목에 있어서 변경이 행해질 수 있다는 것을 인지할 것이다.

발명의 효과

네트워크를 거쳐 시간 및 경비 정보를 획득하고, 그 정보를 프로젝트 회계 시스템 내로 프로세싱하기 위한 시스템이 제공된다.

(57) 청구의 범위

청구항 1.

네트워크를 거쳐 비용 정보를 획득하고, 상기 정보를 프로젝트 회계 시스템내로 프로세싱하기 위한 시스템으로서,

프로젝트에 관련된 시간 및 경비 데이터를 저장하도록 적응된 프로젝트 회계 시스템,

브라우저를 통해 사용자에게 의해 네트워크를 거쳐 액세스될 수 있는 서식(form)으로서 제공되는 사용자 인터페이스 - 상기 서식은 사용자에게 의한 데이터 입력을 위한 필드들과 상기 서식의 전자적 제출을 위한 버튼을 포함함 -, 및

상기 제출된 서식에 포함된 데이터에 기초하여 상기 프로젝트 회계 시스템과의 트랜잭션을 인보크하도록 적응된 서비스 API

를 포함하는 시스템.

청구항 2.

제1항에 있어서,

상기 서식들을 호스트하고 관리된 코드로 작성된 비즈니스 룰에 따라 상기 서식을 사용자에게 제공하도록 적응된 서버를 더 포함하는 시스템.

청구항 3.

제2항에 있어서,

상기 서버는 액티브 서버 페이지 서버이고, 상기 서식은 액티브 서버 페이지 서식인 시스템.

청구항 4.

제1항에 있어서,

상기 웹 기반 서식은 서비스 API로의 호출을 포함하는 시스템.

청구항 5.

제2항에 있어서,

상기 웹 서버는 마이크로소프트 비즈니스 솔루션 비즈니스 포털(Microsoft Business Solutions Business Portal)인 시스템.

청구항 6.

제1항에 있어서,

상기 서비스 API는 .Net 웹 서비스를 사용하여 호출되는 시스템.

청구항 7.

제1항에 있어서,

상기 서비스 API는 리모팅(remoting)을 사용하여 호출되는 시스템.

청구항 8.

제1항에 있어서, 상기 웹 서식은

상기 제출된 웹 기반 서식 내에 포함된 데이터를 프로세싱하기 위한 비즈니스 룰을 인보크하도록 적응된 내장 서비스 API 호출을 더 포함하는 시스템.

청구항 9.

제1항에 있어서,

상기 프로젝트 회계 시스템은 그레이트 플레인 프로젝트 회계 시스템(Great Plains Project Accounting system)인 시스템.

청구항 10.

서식을 통해 회계 데이터베이스로 시간 및 경비 데이터를 획득하는 방법으로서,

서버에서 서식들을 호스트하는 단계 - 각 서식은 사용자가 브라우저를 통해 네트워크를 거쳐 액세스할 수 있고, 또한 사용자 데이터 입력을 위한 데이터 필드들, 사용자에게 의해 완료되면 상기 서식을 제출하도록 적응된 객체, 및 사용자에게 의한 제출 시 관리된 코드로 작성된 비즈니스 룰을 인보크하도록 적응된 내장 서버 컨트롤을 포함함 -, 및

제출된 서식들을 상기 인보크된 비즈니스 룰에 따라 서비스 API로 프로세싱하는 단계

를 포함하며,

상기 서비스 API는 상기 제출된 서식들에 포함된 데이터를 상기 회계 데이터베이스의 엔티티들에 관련시키는 방법.

청구항 11.

제10항에 있어서,

상기 관리된 코드는 공통 언어 런타임 환경(common language runtime environment)으로 작성되는 방법.

청구항 12.

제10항에 있어서, 상기 호스트하는 단계는

복수의 웹 서식을 저장하는 단계 - 상기 복수의 웹 서식 중 적어도 하나는 타임시트 서식임 -, 및

인터넷 브라우저의 윈도우 내에 디스플레이하기 위하여, 상기 복수의 서식 중 적어도 하나를 네트워크를 거쳐 사용자에게 제공하는 단계

를 포함하는 방법.

청구항 13.

제10항에 있어서, 상기 프로세싱하는 단계는

상기 제출된 서식에 포함된 데이터를 상기 비즈니스 룰에 따라 프로세싱하는 단계, 및

상기 프로세싱된 데이터와 상기 비즈니스 룰에 따라 상기 회계 데이터베이스와 상호작용하는 단계

를 포함하는 방법.

청구항 14.

제13항에 있어서,

상기 상호작용하는 단계는, 상기 회계 데이터베이스에 데이터를 저장하는 단계를 포함하는 방법.

청구항 15.

제13항에 있어서,

상기 상호작용하는 단계는, 상기 프로세싱된 데이터에 따라 상기 회계 데이터베이스로 쿼리하는 단계를 포함하는 방법.

청구항 16.

제15항에 있어서,

상기 쿼리하는 단계는 상기 브라우저의 윈도우 내에서 서식에 디스플레이하기 위한 값을 리턴하는 방법.

청구항 17.

네트워크를 거쳐 시간 및 경비 정보를 획득하고, 상기 정보를 회계 시스템 내로 프로세싱하기 위한 시스템으로서,

시간 및 경비 정보를 저장하도록 적응된 회계 시스템,

브라우저를 통한 네트워크를 거쳐 사용자 입력을 위해 적응된 복수의 웹 파트 서식,

상기 사용자 입력을 상기 회계 시스템 내로 프로세싱하기 위하여, 관리된 코드로 작성된 비즈니스 룰을 구현하고 시퀀스화하도록 적응된 서비스 API

를 포함하는 시스템.

청구항 18.

제17항에 있어서,

상기 복수의 웹 파트 서식을 호스트하고, 요청 시 상기 서식들을 사용자들에게 제공하도록 적응된 서버를 더 포함하는 시스템.

청구항 19.

제17항에 있어서,

상기 서비스 API는 상기 회계 시스템과의 트랜잭션을 인보크하는 시스템.

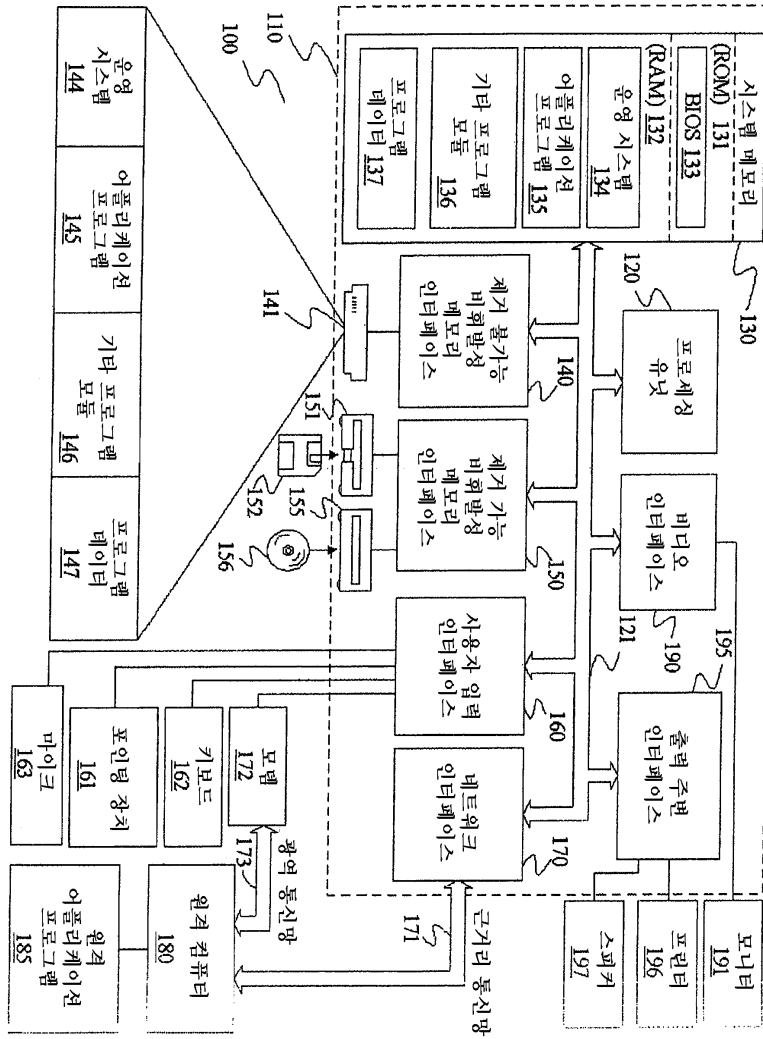
청구항 20.

제17항에 있어서,

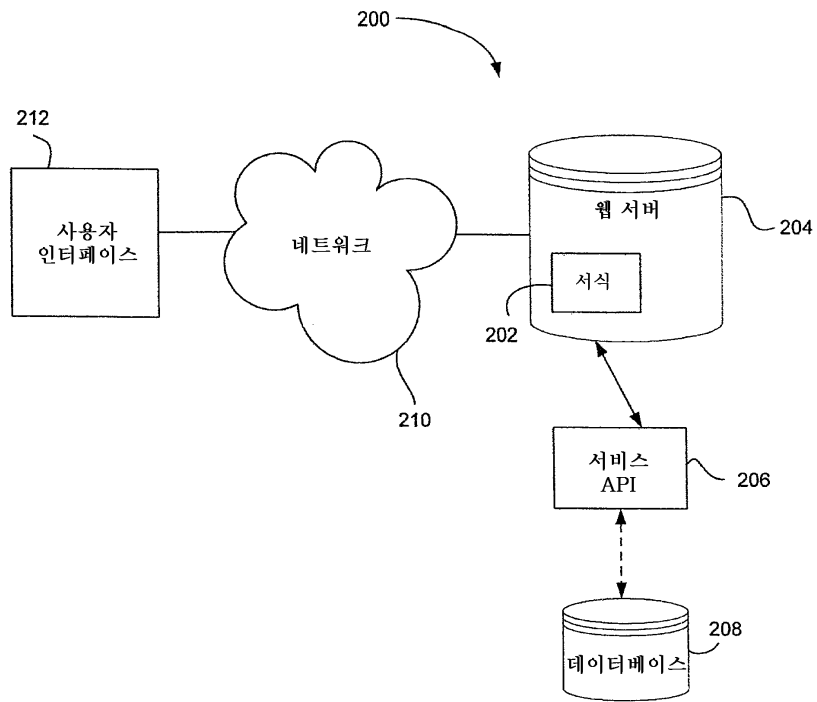
상기 비즈니스 룰은 승인 프로세스에서의 단계들을 포함하는 시스템.

도면

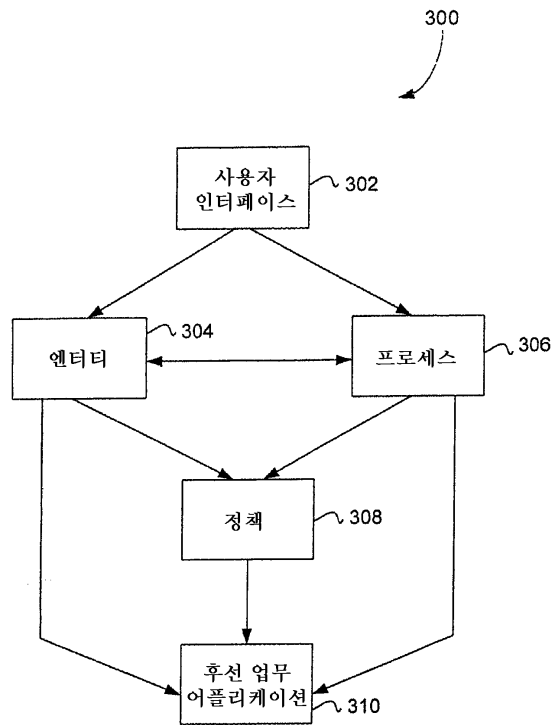
도면1



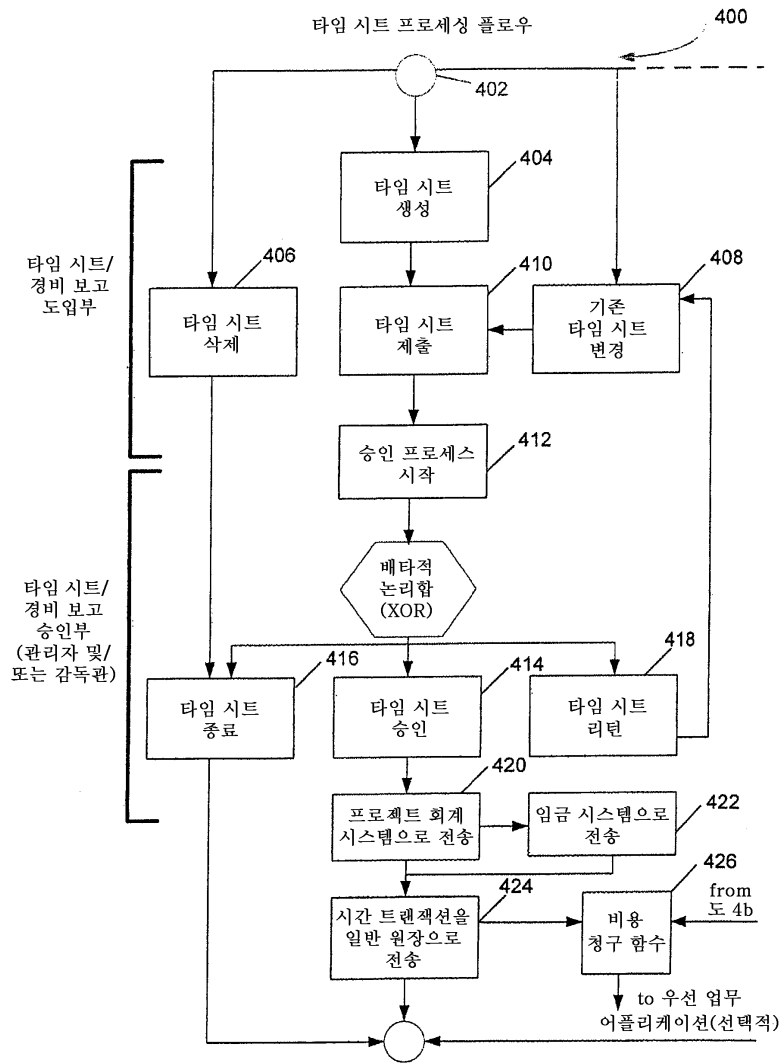
도면2



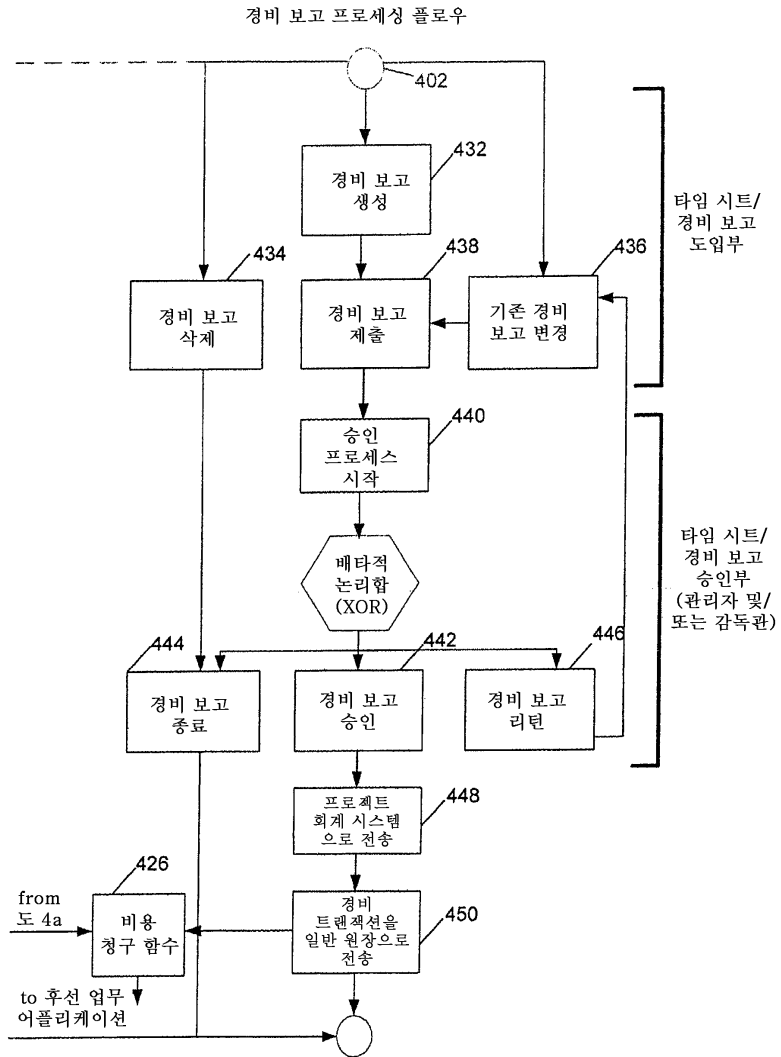
도면3



도면4a



도면4b



도면5

