

논문 2011-1-22

웹 기반 네트워크 근태 관리 시스템 개발

Development of Web-based Network Diligence and Indolence Management System

최우식*, 김병준*, 안병구**

Woo-Sik Choi, Byung-Joon Kim, Beongku An

요 약 기존의 근태관리시스템은 서버와 클라이언트가 분리되어 있지 않고, 데이터가 데이터베이스화 되어있지 않기 때문에 데이터 가공 및 관리 등에 있어서 많은 취약점을 가진다. 본 논문에서는 이러한 문제점들을 보완 해결한 새로운 웹기반 네트워크 근태관리 시스템을 제안 개발한다. 제안 개발 설계된 시스템의 주요한 특징 및 기여도는 다음과 같다. 첫째, 서버와 클라이언트를 분리하고 모든 데이터를 데이터베이스화 한다. 둘째, APM 서버를 구축하여 데이터의 가공 및 관리를 효율적으로 한다. 셋째, 근태관리 시스템의 정책적인 설정을 한다. 넷째, 근태관리 데이터 수집 및 휴무일 집계 업무등 사용자 위주의 서비스를 제공한다. 구현 및 성능평가 결과 제안된 웹기반 네트워크 근태관리 시스템은 효율적으로 근태관리를 지원할 수 있음을 알 수 있으며, 제안된 시스템은 현재 실제적인 현장에 적용하여 사용 중에 있다.

Abstract In recent diligence and indolence management systems, server and client are not separated and data are not convert into data base. Therefore these recent systems have several weaknesses such as data modification and management. In this paper, we propose a new Web-based Network Diligence and Indolence Management system (WNDIM) to solve the weakness and improve the system performance of recent system. The main features and contributions of the proposed system are as follows. First, server and client are separated, and all data are converted into data base, Second, with the construction of APM server the data modification and management are efficiently operated. Third, a political decision of the system is defined. Fourth, the system can efficiently support user-oriented services such as the collection of diligence and indolence data and the sum of off-duty days. From the implementation and performance evaluation of the proposed WNDIM, we can see the system can efficiently support the diligence and indolence management, currently we are using the proposed WNDIM system in the real filed.

Key Words : 근태관리 시스템, Server-Client System, 데이터 베이스, MSSQL

1. 서 론

그림 1에서 보여주고 있는 것처럼 최근에 사용되고 있는 근태관리시스템들은 RF방식의 카드 접촉식

근태관리시스템이다^{[1][2]}. 기존 시스템은 관리자 위주의 접근성이 낮은 불편한 시스템으로 관리자 외에 일반 유저들의 근태 데이터에 관한 접근수요 및 근태자료의 활용과 연동의 필요성이 높아짐에 따라 이를 해결 할 수 있는 새로운 시스템 도입이 필수적이게 되었다. 본 논문에서 제안 제작되는 근태 관리 시스템에서는 사용자 접근이 쉬운 웹기반으로 제작하며, DBMS, 웹 언어(web language), UI 설계, 서버 프로그래밍(server programing)

*준회원, 홍익대학교 컴퓨터정보통신공학과

**중신회원, 홍익대학교 컴퓨터정보통신공학과

접수일자: 2010.12.21 수정일자: 2011.1.21

게재확정일자: 2011.2.11

등 종합적인 기술을 응용하였다.

본 논문은 다음처럼 구성되어 있다. II장은 기존 시스템에 관한 분석을, III장에서는 제안 설계된 웹기반 근태 관리 시스템의 구현 및 설명을 한다. IV장에서는 제안된 시스템의 성능을 검증하고, 마지막으로 V장에서 결론을 맺는다.

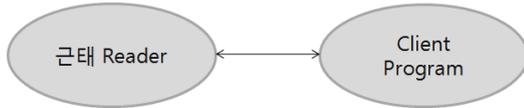


그림 1. 기존 근태관리시스템의 구조도
Fig. 1. The structure of recent diligence and indolence management system

II. 기존 시스템 분석

그림 2에서 설명하고 있는 것처럼 기존 시스템은 RF 근태 기록기로부터 자료를 네트워크를 통하여 관리자 PC(server and client)로 전송하는 구조이다. 따라서 모든 데이터는 관리자 PC에 저장되어 쌓이며 오직 관리자만이 근태 데이터에 접근 가능하다.



그림 2. 기존 근태관리시스템의 하드웨어 구성도
Fig. 2. The hardware structure of the recent diligence and indolence management system

이런 근태관리시스템은 서버와 클라이언트가 통합된 구조로 데이터 안정성에 문제가 생길 수 있다. 또한 오로지 관리자만이 관리 프로그램을 통해 근태 데이터에 접근 할 수 있으며, 다른 사람들이 데이터에 접근하기 어렵다. 이렇게 저장된 근태 데이터는 단순히 프로그램이 설치된 PC의 근접 지역으로 출력하여 버리게(dump) 되기 때문에 다른 용도로 이용 할 수가 없게 된다. 따라서 이러한 점들을 새로운 시스템에서 개선 할 수 있도록 설계를 한다. 그림 3은 기존 시스템의 관리자용 관리 소프트웨어를 보여주고 있다.

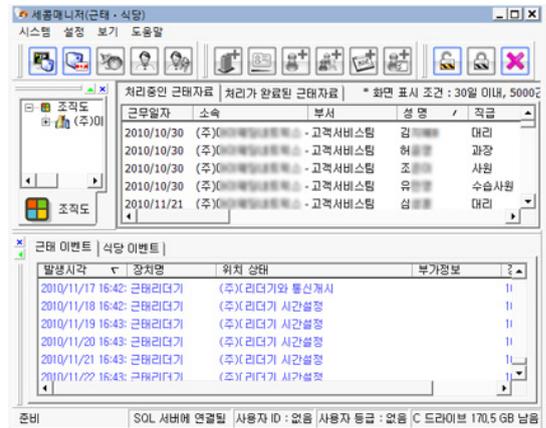


그림 3. 관리자용 관리 소프트웨어
Fig. 3. Administrator's management software

III. 제안된 시스템 설계

기존 시스템의 분석결과를 토대로 새로 제안된 시스템은 다음과 같은 조건을 만족해야 한다.

- 데이터 안정성 및 가공 효율성을 위하여 서버/클라이언트(server/client)의 분리
- 관리자뿐만 아니라 여러 사람들이 근태 기록에 접근 가능
- 편리한 데이터 가공
- 휴무관리 기능과 연동 등 확장성 고려

그림 4는 위에서 설명한 조건들을 만족하여 설계한 본 논문에서 제안한 근태관리 시스템 구성도를 보여주고 있다. 제안된 시스템의 동작 흐름도는 다음과 같다^[3-11].

- Step 1: 근태 리더기로부터 받은 근태 데이터를 MSSQL Server의 Management software를 통해 Raw DB(data base)에 저장한다.
- Step 2: 관리자는 Management software를 통해 Raw 데이터를 수정 할 수 있다.
- Step 3: APM Server의 인터페이스 모듈(interface module)을 통하여 Result DB에 가공된 데이터를 저장한다.
- Step 4: 사용자들은 APM Server의 각종 인터페이스 모듈을 통하여 근태 정보를 확인 할 수 있다. APM Server와 인터페이스 모듈의 자세한 구조는 그림 6. APM Server의 구성도에서 설명하고 있다.

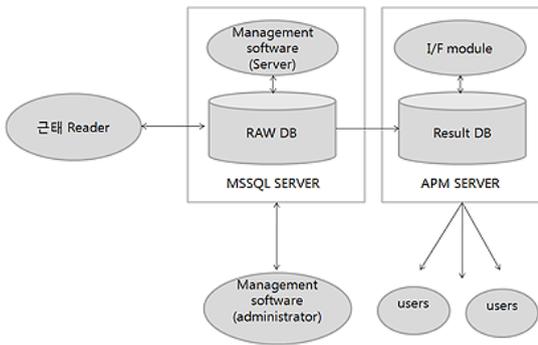


그림 4. 제안된 근태관리시스템의 시스템 구성도
Fig. 4. The structure of improved diligence and indolence management system

제안된 시스템(그림 4) 구성요소들을 좀 더 상세하게 설명하면 다음과 같다^[3-11].

- 근태 Reader: RF방식의 근태처리기 TCP/IP로 연결
- MSSQL Server: RAW DB와 근태 Reader와의 연결 프로그램인 Management software가 설치된 서버이다.
- RAW DB: 근태 Reader로부터의 데이터를 DB화 시켜 RAW 데이터를 작성한다.
- Management software: 근태 Reader의 데이터를 ODBC로 RAW DB에 적제 시켜주는 소프트웨어이다.
- APM Server: RAW 데이터를 기반으로 여러 요구조건에 맞게 기능이 추가된 데이터를 가공하여 웹에 게시하는 apache, php, mysql 서버이다. 자세한 내용 및 구조는 “그림 6. APM 서버의 시스템 구성도”에서 설명하고 있다.
- Result DB: RAW DB로부터 가공된 데이터가 적제 되어 있는 곳이다.
- I/F module: 웹 기반의 근태관리 시스템의 인터페이스 모듈들이다.
- Management software: 기존의 관리자 소프트웨어는 MSSQL에 연결하여 사용이 가능하다.
- USER: 일반사용자들은 웹페이지에 접속하여 자신의 근태현황을 확인 할 수 있다.

1. Server-Client System 분리

그림 5는 서버-클라이언트 기반의 제안된 시스템을 설명한다. 제안된 시스템을 구축하기에 앞서 기존의 시

스템을 서버-클라이언트 시스템으로 재구축 하여 RAW 데이터를 확보해야 한다. 그러기 위해서 MSSQL 기반의 윈도우 서버가 구축되어야 하며, MSSQL DB와 업체에서 제공되는 서버관리 프로그램이 설치되어야 한다^[3-11].

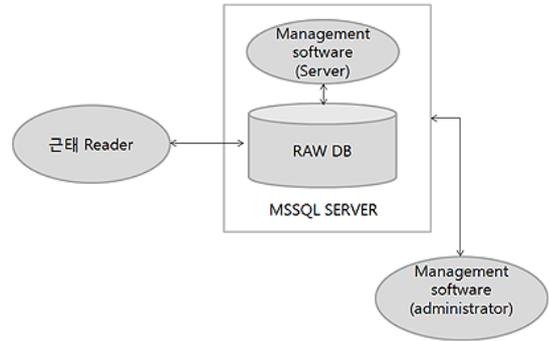


그림 5. 서버-클라이언트 기반의 시스템
Fig. 5. Server-Client based system

윈도우 서버는 윈도우 2008을 사용하여 구축하며 DB는 MSSQL STD 2005을 이용한다. Management software는 근태 Reader와 연동하기 위한 미들웨어로 ODBC로 연결한다.

2. APM Server 구축

서버-클라이언트 시스템으로 분리가 성공하면 본격적으로 APM 서버를 구축한다. 그림6은 APM 서버의 시스템 구성도를 설명한다. APM에서의 동작 흐름도는 다음과 같다^[3-11].

- Step 1: MSSQL의 server programe인 cron_attend가 서버 안에서 스케줄링 되면서 작동한다. cron tab을 통하여 매 5분 동안 동작하도록 스케줄링 설정.
- Step 2: cron_attend에서는 MSSQL 서버로부터 데이터를 가져와서 Result DB에 적제하는 동시에, 주어진 정책에 따라 지각 여부판단 등 다른 I/F 모듈과 통신하면서 처리하게 된다.
- Step 3: 사용자들은 일별 지각 현황 페이지인 list와 월별 지각 현황 페이지인 list_Jate, 자신의 휴무 및 지각 여부를 알 수 있는 attendance 페이지에 접근 할 수 있다.
- Step 4: Step 3에서 설명한 각종 페이지 들은 관리자/사용자의 기능 및 화면이 다를 수 있다.

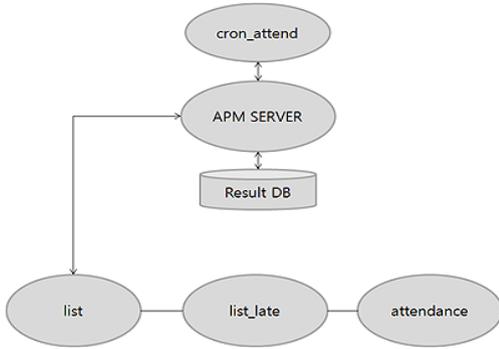


그림 6. APM 서버의 시스템 구성도
Fig. 6. The system structure of APM server

APM Server 및 I/F 모듈의 기능을 좀 더 상세하게 설명하면 다음과 같다.

- RAW DB: 근태 관리기로부터 쌓이게 되는 근태 자료가 있는 DB.
- Result DB: RAW DB로부터 가져온 정보와 각 사용자의 추가로 가공된 데이터를 가지고 있는 DB.
- Cron_attend: 근태관리시스템의 core 프로그램으로 cron program. RAW DB로부터 Result DB에 5분마다 새로운 근태 자료를 및 추가로 가공된 업데이트하며 설정된 지각/휴무 정책을 반영하여 off_day, off_day_out table에 기록.
- List: 일별 근태 현황 보기 페이지로 각 담당자의 일별 출근시간, 퇴근시간 이달 지각횟수를 알 수 있음. 또한 관리자를 위한 근무시간, 18:30분 이전 퇴근한 사람 목록을 제공.
- List_late: 월 별 누적 지각 현황 보기 페이지로 각 담당자의 이번 달 지각 횟수, 카드 미체크 횟수가 표시되며 관리자를 위한 총 근무시간 기능을 제공.

3. 근태관리시스템의 정책 설정

근태관리시스템의 정책은 cron_attend에 프로그램 되어있다. 설정된 정책에 따라 지각 및 휴무를 판단하여 자동으로 반영하게 된다.

- 정책 1: 임직원들은 일반출근(09:30분까지), 야근(10시까지 출근) 오전휴무, (13:30분까지 출근 가능) 오후휴무(15:30분 퇴근), 휴일을 선택 할 수 있으며 해당 값은 이미 구축하여 사용 중인 attendance 테이블에 각각의 임직원 별로 날짜마다 들어있다.

- 정책 2: 임직원 별로 attendance 테이블과 비교하여 지각유무를 체크한다. 예를 들어 일반 출근하는 날인데도 불구하고 09:30분 이후에 카드를 찍게 되면 지각처리가 된다. 만약 야근을 체크했다면 10시 이전까지는 정상 출근 처리, 그 이후는 지각으로 판단하여 처리한다.
- 정책 3: 휴무일이 아닌 일반 출근 일에 근태기록이 없으면 자동으로 휴무로 처리한다.
- 정책 4: 퇴근 시에만 근태 데이터가 있고 출근 시 근태 데이터가 존재하지 않는다면 지각처리를 한다.
- 정책 5: 정상적인 퇴근시간 전에 퇴근한다면 관리자 페이지에 '퇴근시간 이전 퇴근한 사람목록'에 나타내게 된다.
- 정책 6: 출/퇴근 데이터가 정상적으로 존재 할 때 근무시간이 계산되며 둘 중 하나가 존재하지 않는다면 근무시간이 계산되지 않는다.

4. 일별 근태현황 보기 UI설계

그림 7에서 설명하고 있는 것처럼 실질적으로 사용자가 접속하여 보게 되는 웹페이지로 임직원들의 근태데이터가 있다. 해당 페이지는 다음과 같은 기능을 가진다.

- 년도/월/일 별로 근태 데이터를 조회 가능해야 한다.
- 출력되는 정보는 번호/이름/ID/출근시간/퇴근시간/이번 달 지각횟수이다.
- 모든 필드는 보기 편하게 순서정리(sorting)가 가능해야 한다.
- 관리자만 볼 수 있는 각각의 근무시간, 퇴근시간 이전에 퇴근한 사람 목록을 제공한다.

No	이름	ID	출근시간	퇴근시간	이번달 지각횟수	근무시간
1	강	kangsy	2010-11-13 08:59:00	2010-11-13 18:41:00	0	09:42
2	강	jiso	2010-11-13 09:12:00	2010-11-13 16:03:00	0	06:51
3	강	jsk	2010-11-13 09:22:00	2010-11-13 16:10:00	0	06:48
4	고	koyk	2010-11-13 13:39:00	2010-11-13 20:42:00	1	07:03
5	공	sooyoung	2010-11-13 09:28:00	2010-11-13 19:03:00	1	09:35
6	곽	swkwak	2010-11-13 09:21:00	2010-11-13 19:43:00	3	10:22
7	구	kush	2010-11-13 09:10:00	2010-11-13 18:59:00	0	09:49
8	권	yuyu	2010-11-13 09:19:00	2010-11-13 20:44:00	1	11:25
9	김	kkah	2010-11-13 09:20:00		미체크	2
10	김	nezmi91	2010-11-13 09:32:00	2010-11-13 22:19:00	4	12:47
11	김	njkim	2010-11-13 09:23:00	2010-11-13 18:30:00	0	09:07

그림 7. 일별 근태현황 웹페이지 화면
Fig. 7. Webpage view of daily diligence and indolence

5. 월별 누적 지각보기 시설

그림 8에서 설명하고 있는 것처럼 해당 페이지에서는 월별로 누적된 지각 횟수와 카드 리더기에 체크하지 않은 경우인 미 체크로 인한 지각 횟수를 확인 할 수 있으며 관리자만 볼 수 있는 총 근무시간과 퇴근시간 이전 퇴근한 사람들의 목록을 볼 수 있다.

No	이름	ID	미반달 지각횟수	미체크 횟수	총 근무시간
1	강	kangsy	1	1	252.7
2	강	kej	3	0	183.2
3	강	jiso	0	0	195.55
4	강	jsk	0	1	0
5	고	koyk	2	1	175.42
6	공	sooyoung	1	0	206.48
7	권	swkwak	3	0	244.47
8	구	kush	1	0	255.23
9	권	yuyu	0	1	231.32
10	김	kkah	1	0	133.3
11	김	nezmi91	2	1	269.57

그림 8. 월별 누적 지각보기 웹페이지 화면
Fig. 8. Webpage view of monthly late check-in/out

IV. 제안된 시스템의 성능평가

제안 구현된 시스템의 성능평가는 다음의 과정으로 검증한다.

- 1) 정상적으로 RAW DB에 적재여부
- 2) cron program의 정상 작동 여부
- 3) 데이터의 정상적인 입력/업데이트(insert/update)
- 4) 설정한 정책대로 작동여부

1. 정상적으로 RAW DB에 적재여부 확인

그림 9는 출근 시 출입카드를 근태리더기에 접촉시켜서 출근 확인할 수 있음을 보여준다.



그림 9. 출입카드로 근태리더기에 출근 확인
Fig. 9. using card to confirm attendance

그림 10은 #MSSQL 서버에 있는 관리프로그램 (management program)에서 출근 시간을 확인할 수 있음을 보여준다.

사원명	출근시간	퇴근시간
김병준	2010/10/23 13:16	2010/10/23 21:59

그림 10. 관리 프로그램에서 출근 시간을 확인
Fig. 10. Check attendance time in management program

그림 11은 MSSQL 서버에 있는 Raw DB에서 출근 시간을 확인할 수 있음을 보여준다.

```

SELECT workDate, name, wSTime
FROM tWORKHISTORY
WHERE (workDate = '20101023') AND (name = '김병준')
    
```

workDate	name	wSTime
20101023	김병준	20101023131600
NULL	NULL	NULL

그림 11. Raw DB에서 출근 시간을 확인
Fig. 11. Check attendance time in RAW DB

2. Cron program의 정상 작동 여부

그림 12는 APM 서버의 결과 DB에 입력여부를 확인할 수 있음을 보여준다.

today	name	id	check_in	check_out
20101023	김병준	bjkm	2010-10-23 13:16:00	2010-10-23 21:59:00

그림 12. 결과 DB에서 데이터 확인
Fig. 12. Check data in Result DB

그림 13은 APM 서버의 일별 근태 현황에서 데이터 입력여부를 확인할 수 있음을 보여준다.

12	김병준	bjkim	2010-10-23 13:16:00	2
13	김	sykim	2010-10-23 12:10:00	2
14	김	suekim	2010-10-23 09:27:00	2
15	김	shkim	2010-10-23 10:10:00	2
16	김	ksina79	2010-10-23 09:21:00	2
17	김	ahrum1	2010-10-23 09:23:00	2

그림 13. 일별 근태 현황에서 데이터 확인
Fig. 13. Check data in daily check in/out

3. 데이터의 정상적인 입력/업데이트 (insert/update)

그림 14는 cron_attend 프로그램의 데이터 입력을 확인 할 수 있음을 보여준다.

```
if (${cron_attend_comapre_row[name]} == '') ( //cron_attend에서 데이터가 없다면
    $sql_l="INSERT INTO cron_attend ( no, today, name,id,
    scheduleName,check_in_stamp,check_out_stamp, check_in, check_out,
    insert_time) VALUES ('', '', $today, '', '', $secom_row[name],'',
    '', $id_row[id],'', '', $secom_row[scheduleName],'',
    '', $check_in_stamp, '', '', $check_in_date, '', '', $insert_time.'');
    $rs=mysql_db_query('center', $sql_l, $connection);
    $echo '#insert 됐음#'; //cron_attend table에 insert 시킴
```

그림 14. cron_attend 프로그램의 데이터 입력 및 입력 확인 부분
Fig. 14. Check data insert in cron_attend

그림 15는 cron_attend 프로그램의 데이터 업데이트를 확인 할 수 있음을 보여준다.

```
if (${check_out_stamp} == ${check_out_null}) {
    $echo 'update 관계 없음.'; //퇴근 데이터가 없으면 update안함
```

그림 15. cron_attend 프로그램의 데이터 업데이트 및 업데이트 확인 부분
Fig. 15. Check data update in cron_attend

4. 설정한 정책대로 작동여부

그림 16 -그림 20은 설정한 정책대로 시스템이 작동하는지를 검증하기 위해서 일별 근태 현황에서 다섯 가지의 기능들을 다음처럼 평가 확인 하였다.

그림 16은 일반출근(09:30분까지)일 경우 지각을 했을 때 지각 표시를 보여준다. 즉, 일반 출근시간(09:30)이 지났으므로 자동으로 지각처리 됨을 확인 하였다.

71 윤○○○ yk0078 2010-11-05 09:31:00						
개인	구분	1 Mon	2 Tue	3 Wed	4 Thu	5 Fri
	일반		일			
윤○○○	야간					야
	지각					
윤○○○	일반		일			
	야간					
	지각		지			지

그림 16. 일반 출근일 경우 09:30분 이후 출근은 지각 처리
Fig. 16 The absence process of attendance after 09:30 in regular attendance

그림 17은 야근(10:00분까지)일 경우 지각을 했을 때 지각 표시를 보여준다. 즉, 야근 출근시간(10:00)이 지났으므로 자동으로 지각처리 확인 하였다.

22 김○○○ dmsruds 2010-11-08 10:31:00									
개인	구분	1 Mon	2 Tue	3 Wed	4 Thu	5 Fri	6 Sat	7 Sun	8 Mon
	일반			일			일		
김○○○	야간								야
	지각								지

그림 17. 야근 출근일 경우 10:00분 이후 출근은 지각 처리
Fig. 17 The absence process of attendance after 10:00 in night attendance

그림 18은 오전휴무(13:30분까지)일 경우 지각을 했을 때 지각 표시를 보여준다. 즉, 오전휴무 출근시간(13:30)이 지났으므로 자동으로 지각처리 됨을 확인 할 수 있다.

72 임수진 sjim 2010-11-06 13:55:00							
개인	구분	1 Mon	2 Tue	3 Wed	4 Thu	5 Fri	6 Sat
	일반		일	오			
임수진	야간						
	지각						
임수진	일반		일	오			오
	야간	야			야		
	지각						지

그림 18. 오전휴무 출근일 경우 13:30분 이후 출근은 지각 처리
Fig. 18 The absence process of attendance after 13:30 in off-duty of AM.

그림 19는 근태기록이 퇴근 때만 있을 경우 자동으로 지각 처리(출근 시간이 언제인지 확인 할 수 없으므로)를 할 수 있음을 보여준다. 즉, 출근시간 데이터가 존재하지 않으므로 자동으로 지각처리 됨을 확인 할 수 있다.

2010년 11월 6일 보기							
No	이름	ID	출근시간	퇴근시간			
21	김○○○	uaroomb	미체크	2010-11-06 18:34:00			
개인	구분	1 Mon	2 Tue	3 Wed	4 Thu	5 Fri	6 Sat
	일반		일				오
김○○○	야간						
	지각						지

그림 19. 출근시간 데이터가 존재하지 않으면 자동으로 지각 처리
Fig. 19 The absence process in the case of no attendance time data

그림 20은 출근 시간 및 퇴근시간 데이터가 있다면 자동으로 근무시간 계산 둘 중 하나가 없다면 미체크로 표시 할을 보여준다.

No	이름	ID	출근시간	퇴근시간	이번달 지각횟수	근무시간
21	김	uaroom	미체크	2010-11-06 18:34:00	1	미체크
61	이	swlee	미체크	2010-11-06 20:19:00	1	미체크
90	지	sage	미체크	2010-11-06 18:51:00	2	미체크
63	이	sslee	2010-11-06 07:52:00	미체크	0	미체크
46	심	simsh	2010-11-06 08:10:00	2010-11-06 19:39:00	2	11:29

그림 20. 출근 및 퇴근시간 데이터에 따른 근무시간 자동 계산

Fig. 20 The automatic calculation of working time according to the check-in/out data

V. 결 론

본 논문에서는 웹기반 네트워크 근태관리 시스템(WNDIM)을 제안 개발 하였다. 본 연구에서 제안 구현한 시스템의 범위는 DBMS와 서버-클라이언트 시스템, 웹 언어 등 인터넷 기반의 웹서버 시스템을 구축과정부터 실제 서비스 구현까지의 총 과정이 모두 포함되며, 이러한 내용을 직접 연구함으로써 종합적인 기술을 응용하여 축적하는 귀중한 기회가 되었다. 이렇게 제안 구현된 WNDIM 시스템은 기존시스템과는 차별화된 기능으로 사용자의 편리성과 함께 다양한 추가 기능을 사용 할 수 있었다. 현재는 내부 사내 정책에 맞게 제작되었지만 웹 기반으로 제작하였기 때문에 현재 구현된 기능 외에도 사용자의 요구에 의해 얼마든지 다양한 모듈을 제작하여 사용화 할 수 있는 확장성을 지니고 있다. 기존 시스템이 관리자 위주의 단방향의 근태확인만 할 수 있는 시스템이라면 개선된 시스템에서는 관리자뿐만 아니라 유저들도 접속하여 확인 할 수 있는 양방향 시스템이며 단순히 근태 확인에만 그치지 않고 정책에 따른 자동 휴무관리 기능까지 가지고 있는 웹기반의 사원관리 해결방안 시스템이라고 할 수 있다. 현재 제안 개발된 WNDIM 시스템은 현장에 적용하여 사용 중에 있으며, 그 기능을 보완 향상 중에 있다.

참고문헌

- [1] Reiko IWAI, Hiroyuki YOSHIMURA, "Security of registration data of fingerprint image with a server by use of fractional Fourier transform," Proc. of ICSP2008, 2008.
- [2] 이종상, "RFID 기술을 적용한 출입통제관리시스템 구축사례연구," 연세대학교 석사학위논문, 2006.
- [3] Minera, Francisco Jose, Mp Ediciones, Corp Php Y My SQL Php & Mysql (Paperback), 2006.
- [4] JOSEPH M.HELLERSTEIN, READINGS IN DATABASE SYSTEMS-4/E, 2009.
- [5] APRESS, SQL SERVER CE DATABASE DEVELOPMENT WITH THE. NET CO, 2009.
- [6] 루크 웰링, 로라 톰슨 공저, 류정욱,김소희 공역, 성공적인 웹 프로그래밍 PHP와 MySQL, 2009.
- [7] 폴드보이 저, 김형훈 역, MySQL : 한국어판, MySQL의 사용, 관리, 프로그래밍을 위한 완벽 가이드 (4판), 2009..
- [8] 손광수, 데이터베이스 관리와 실습 : MS SQL 서버 IT COOKBOOK, 2003.
- [9] 이승혁, PHP5 웹 프로그래밍 가이드 WEB PROGRAMMING GUIDE, 2006.
- [10] 래리 올만, PHP 6 & MySQL 5 : 다이내믹한 웹 사이트 들기, 2010.
- [11] 홍의경, 데이터베이스 배움터 -MS SQL SERVER 기반, 2007.

※ 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(2010-0022162).

저자 소개

최 우 식(준회원)



- 2003년~현재 : 홍익대학교 컴퓨터정보통신학과 재학
- <관심분야 : web 2.0, web UI, web-programming, graphic>

김 병 준(준회원)



- 2004년~현재: 홍익대학교 컴퓨터정보통신학과 재학
- <관심분야 : PHP, ASP, JSP 등 웹언어, Database 설계, Server 구성, Storage server 구성>

안 병 구(중신회원)



- 1988년: 경북대학교 전자공학과 (BS)
 - 1996년: (미)Polytechnic University, Dept. of Computer and Electrical Eng., Brooklyn, New York, USA (MS).
 - 2002년: (미)New Jersey Institute of Technology(NJIT), Dept. of Computer and Electrical Eng., New Jersey, USA.(Ph.D)
 - 1989년-1994년: 포항산업과학기술연구원(RIST), 선임연구원
 - 1997년-2002년: Lecturer & RA, New Jersey Institute of Technology(NJIT), USA.
 - 2003년- 현재: 홍익대학교 컴퓨터정보통신공학과 교수
 - 2005년-2010년:Marquis Who's Who in Science and Engineering was listed. (세계과학기술인명사전 등재)
 - 2006년-2010년:Marquis Who's Who in the World was listed. (세계인명사전 등재)
- <주관심분야: Wireless Networks, Ad-hoc & Sensor Networks, Multicast Routing, QoS Routing, Cross-Layer Technology, Cooperative Communication, Network Coding, Bioinformatics, VLC>