

Web Site에서의 교내 기자재고장처리 확인을 위한 시스템 개발

조규철*, 전세연^o

^o인하공업전문대학 컴퓨터정보과

e-mail: kccho@inhac.ac.kr*, tktktk654@naver.com^o

Development of a system for confirming the failure of equipment in Web Site

Cho Kyu Cheol*, Jeon Se Yeon^o

^oDept. of Computer Science, Inha Technical College

● 요약 ●

교내 기자재가 고장이 나면 사용자는 수리하는 담당자에게 직접 연락을 하여 조취를 취한다. 고장수리 담당자의 처리 여부에 대한 확인은 유선으로 상호간에 확인하는 방법뿐이다. 본 연구를 통해 고장 난 기자재를 처리하고 싶은 사용자를 위해 실시간으로 처리 현황을 확인 할 수 있는 인터페이스와 기능을 지원하는 시스템을 개발하였다. 관리자는 해당 분류에 맞게 처리하는 담당자를 구별해 시스템에 글을 게시하고, 담당자는 담당 부서 게시물만 열람할 수 있도록 하여 빠른 일처리를 지원할 수 있는 기능을 중점으로 개발하였다. 또한 하나의 기자재마다 책임자를 구별하고 처리 완료된 기자재의 종류의 기록과 통계를 사용자와 담당자가 보기 어려운 단위나 수치보다는 시각적인 효과를 받기 위한 그래프의 사용과 색상을 통하여 알림을 전해준다.

키워드: 고장 기자재(Failure equipment), Web based System

I. 서론

교내에서 기자재가 고장이 나면 해당 담당자에 직접 연락을 하고 처리가 완료되면 해당 사용자에게 다시 연락을 함으로써 업무를 마무리한다.

사용자는 실시간으로 처리상황을 알 수 없고, 담당자는 사용자에게 받은 연락이 잘못된 정보라면 처리하는 과정에서 불편함을 가지고 있고, 학교 측에서는 처리되는 기자재의 기록과 통계에 잘못된 데이터가 입력되어 문제점을 가지고 있다. 본 연구는 이에 대해 사용자들의 불편함을 느끼고 이를 개선하고자 개발하게 되었다.

II. 관련 사례

모든 회사는 고객센터가 존재한다. 특히 전화로 연결이 되는 고객센터는 어떤 분류의 문의가 들어왔는지 자동응답기로 구별한다. 즉, 분류를 결정하는 내비게이션 기능을 가지고 있다.

본 연구는 이를 참고하여 고장 난 기자재의 분류를 나눈다. 고장 난 기자재를 처리하고 싶은 사용자는 내비게이션기능을 가진 관리자에게 전화를 한다. 관리자는 해당 분류에 맞게 처리하는 담당자를 구별해 웹사이트에 글을 작성한다. 이 때, 내선번호를 기준으로 적힌 Excel 정보를 파싱을 해서 비동기적으로 자동입력하게 하여 오류 없는 데이터 입력의 받는다.

또한 서비스를 제공하는 시스템에서는 A/S를 처리현황을 기록한다. 대표적인 예로 인터넷 쇼핑몰 11번가에서는 고객별, 제품별 등 처리현황을 CRM(Customer Relationship Management) 프로그램에서 표 형식으로 보여준다. 하지만 앞서 언급한 분류별 내용을 한눈에 볼 수 없고, 시각적인 효과에서 많이 불편한 점이 있다.

이를 참고하여 고장 난 기자재의 작업처리가 완료가 되면 일반 사용자도 알아보기 쉽도록 차트를 이용한 기능을 가지고 있다. 또한 처리 완료가 되면 자동으로 기록과 통계가 합산되는 점에서 데이터의 무결성을 가진다.

본 연구는 High-Chart[1]를 사용하여 진행하며 MVC(Model View Control)의 이해를 기반으로 하고 있다. 동적으로 변하는 데이터를 다루기 위하여 MVC 의 형태의 이해를 해야 실시간으로 변하는 데이터를 컨트롤 할 수 있다.

III. 서비스 기반 웹사이트에 적용한 MVC 구조

Model-View-Control(MVC) 구조는 기본적으로 세 부분으로 나누어져 있으며 입출력을 담당하는 뷰는 클라이언트에만 존재하고 나머지 컨트롤러와 모델만 기능에 따라 나누면 되기 때문에 서비스 기반 웹사이트를 설계함에 있어서 적합하다. 여기서 사용된 MVC구조는 Thin-Client MVC[2] 구조이다.

그림 1은 Thin-Client MVC 구조를 도식화한 것으로 사용자에게 정보를 입력하고 출력하는 뷰는 웹사이트 클라이언트에 존재하고 컨트롤러와 모델은 서버에서 서비스로 제공된다. 이때, 컨트롤러가 웹사이트 클라이언트에서 분리됨으로써 복잡한 연산 시에 클라이언트의 낮은 연산 능력 때문에 발생하는 문제점을 극복할 수 있다. 또한 모델의 분리는 부족한 데이터 저장 공간을 확보하고 컴퓨터 디바이스 분실 및 파손으로 인한 데이터 손실을 막을 수 있다.

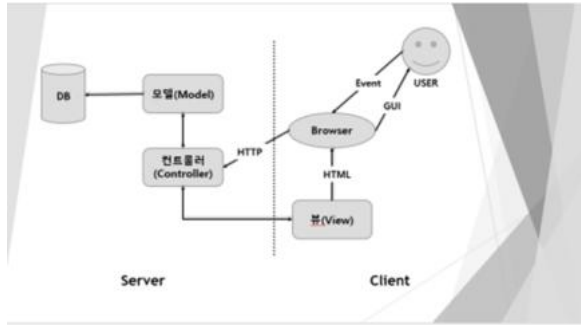


Fig. 1. Thin-Client MVC

기자재가 처리가 되면 그림 2와 같이 데이터베이스에 들어있는 값을 가져와 차트에 표기하기 위해서 High-Chart에 코드를 작성한다.

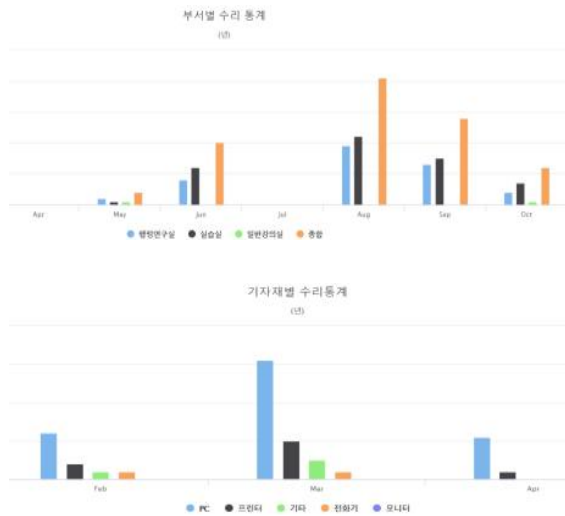


Fig. 2. Chart representation using MVC

IV. Conclusions

As-Is 모델에서는 고장 난 기자재를 처리하는 담당자들은 잘못된 전달받은 정보로 처리하는 과정에서, 사용자측에서는 처리현황을 알 수 없기 때문에 불편한 점이 있었다. 또한 학교 측에서는 잘못된 데이터를 입력하는 통계정보를 신뢰할 수 없었다는 문제점도 가지고 있었다.

본 연구에서 As-Is 모델의 문제점을 그림 3과 같이 개선한 To-Be 모델이 더 다양하고 많은 교내 시스템을 개선할 수 있을 것으로 예상된다.

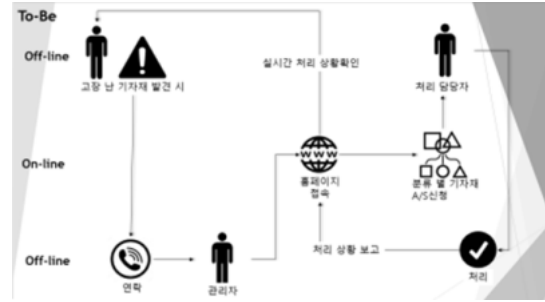


Fig. 3. To-Be of Research & Development

REFERENCES

- [1] HighChart, <https://www.highcharts.com/demo>
- [2] Thin-Client MVC <https://www.kci.go.kr/kciportal/ci/sereArticleSearch/ciSereArtiOrteView.kci?sereArticleSearchBean.artiId=ART001494347>