

웹기반 품질기능전개(QFD)용 소프트웨어 개발 †

이정환* · 이승훈* · 윤석동**

*동의대학교 정보산업공학과

**세이프티넷

Development of a Web-based QFD Software

Jung-Hwan Lee* · Seung-Hoon Lee* · Seuk-Dong Yoon**

*Dept. of Information & Industrial Engineering, Dongeui University

**SafetyNet

QFD(Quality Function Deployment) is a system for translating specific customer needs into detailed product requirements which are deployed throughout design, production, marketing and support operations. QFD is a team tool. However, most of commercial QFD softwares are standalone and therefore offer limited support for teamwork. In this paper, we developed a web-based QFD software to overcome many of the limitations that standalone systems suffer from. That is, this software was developed for synchronizing collaboration between team members distributed not only geographically but also over time. This software was developed by using Visual studio .NET and Microsoft COM+(CBD) technique and can be implemented on the web browser.

Keywords : QFD, Web-based, Synchronous Collaboration, CBD⁷⁾

1. 서 론

QFD(Quality Function Deployment : QFD)는 신제품 개념정립, 설계, 부품계획, 공정계획, 그리고 생산계획과 판매까지 모든 단계를 통해 고객의 요구가 최종 제품과 서비스에 충실히 반영되도록 하여 고객의 만족도를 극대화시키는 품질경영의 한 기법이다. QFD의 기본 개념은 고객의 요구조건을 제품의 기술특성으로 변환하고, 이를 다시 부품특성과 공정특성, 그리고 생산에서의 구체적인 사양과 활동으로까지 변환하는 것이다²⁾. QFD의 전체적인 목적은 신제품의 개발기간을 단축하고 동시에 제품의 품질을 향상시키는 것이며, 이런 목적을 달성하기 위하여 신상품 개발의 초기단계부터 마케팅부서, 기술부서 및 생산 부서가 서로 밀접하게 협력해야 한다. 즉, QFD는 팀을 이루어 노력하는 복잡한 활동이다.

QFD 프로젝트를 수행함에 있어 나타날 수 있는 문제점은 시간과 장소의 제약으로 프로젝트 팀원들이 같은 시간, 같은 장소에 있지 않으면 프로젝트를 진행하기가 어렵다는 것이다. 예로 다국적 기업에서 신제품을 개발한다고 하면 다음을 생각하여 볼 수 있다. QFD 프로젝트의 시작은 소비자의 요구에 대한 조사로부터 시작한다. 소비자의 요구를 제품에 반영하기 위해 소비자 요구조사 다음 단계의 작업인 소비자의 요구에 대응하는 품질특성을 추출한다. 품질특성의 추출은 아무나 할 수 있는 작업이 아니다. 해당 제품에 대한 전문가나 기술자들에 의해 수행되는 작업이다. 만약 기술자들이 멀리 떨어진 다른 국가(다국적 기업의 본사)에 있다면 QFD 프로젝트 진행을 위한 기술자들의 이동과 시간에 따른 비용이 발생하게 된다.

상용화된 QFD 소프트웨어는 외국제품으로는 QFD Designer

† 이 논문은 2003학년도 동의대학교 자체 학술연구구성비의 지원을 받아 작성되었음.

(Qualisoft/Fulfillment Services, <http://www.qualisoft.com>), QFD Scope(Integrated Quality Dynamics, <http://www.iqd.com>), QFD/Capture (International TechneGroup, <http://www.qfdcapture.com>), QFD 2000(Total Quality Software, <http://www.qfd200.co.uk>), Qualica QFD(qualica Software, <http://www.qualica.de>), VOCALYST(Applied Marketing Science, <http://ams-inc.com>) 등이 대표적이다. 이들 소프트웨어에 관한 기능과 특징을 비교·평가한 내용은 Herzwurm 외 2인(2003)을 참조하기 바란다. 그리고 국내 제품으로는 QFD 21C(유니보스, <http://www.uniboss.com>)가 있으며, QFD 21C, QFD 2000, Qualica QFD를 비교 분석한 내용은 이상복과 신동철(2002)의 15장을 참조하기 바란다. 최근 이희석 외 2인(2005)은 비용을 고려한 품질기능전개에 대한 방법론을 제시하고 이를 소프트웨어로 개발하였다. 상기 소프트웨어의 대부분은 stand-alone 방식이다. 따라서 상기 소프트웨어로는 QFD 프로젝트를 수행함에 있어서 프로젝트 팀원들이 같은 시간, 같은 장소에 있지 않으면 프로젝트를 진행하기가 어렵다.

본 논문에서는 이러한 시간과 장소의 제약을 보완하고 동기적 협업(Synchronous Collaboration)이 가능한 웹기반 QFD 소프트웨어를 개발하고자 한다. 최근 웹기반 QFD 소프트웨어 개발에 관한 연구로는 Huang과 Mak(2002)이 있다. 이 연구에서는 Microsoft ActiveX 기술을 이용하여 웹기반 QFD System Prototype을 제안하였다. 이 시스템은 3-Tiered Client-Sever 구조이며 개발에 사용된 원격지간 작업 공유 매커니즘은 단순 투표 시스템을 사용하였다. 또한 시스템과 별도로 온라인 메시지 매커니즘을 활용하여 작업 수행시 의사소통을 할 것을 제안하였다. 하지만 이 시스템은 하나의 의견이 제안되면 다른 사용자 전체의 동의가 있기 전까지 기다려야 하며 의견 조율을 원활하게 수행할 수가 없다. 또한 단순 투표 시스템은 실시간으로 작업내용이 적용되기는 어려우며 데이터 공유 기능 이외의 작업을 수행하는데 어려움이 있다. 구조적인 측면에서 클라이언트/서버 구조일 경우에는 관리 면에서 어려움이 있으며 응용프로그램의 사소한 변화에도 재배포하는 불편함이 있다. 그리고 모든 비즈니스 로직을 클라이언트 단독으로 처리할 경우 복잡한 로직을 실행하려면 클라이언트의 과부하가 발생한다. 개발 기술 측면에서 ActiveX는 윈도우 환경(Internet Explore)에서만 실행가능하며 개발 언어 종속적이며 추후 확장(XML지원)과 관련하여 제약이 있다.

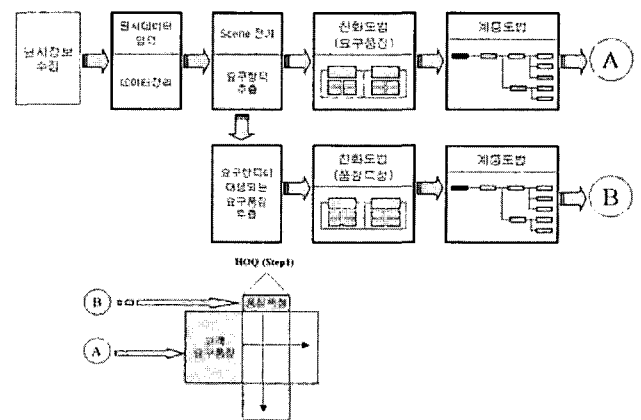
이에 본 논문에서는 원격지에 떨어진 사용자들이 시간과 공간에 구애 받지 않고 실시간으로 원하는 일을 동시에 진행할 수 있는 소프트웨어 설계 및 시스템을 개발하고자 하며, 그 구조는 N-Tiers 분산 환경으로 응용 프로그램을 비즈니스 로직과 사용자 인터페이스를 클라이언트와 서버에 적절하게 분산함으로써 관리, 보안, 과

부하 문제를 해결하려 하고 .Net기술과 COM+ 기술을 사용하여 ActiveX 단점을 보완하고자 한다. 또한 응용프로그램 공유 매커니즘을 이용하여 실시간으로 원격지간 작업을 공유하도록 한다.

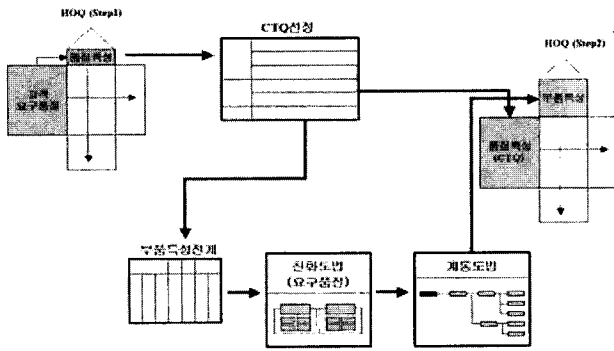
2 QFD 전개 및 지원도구

2.1 QFD 전개

QFD의 작성단계에 대한 정형적인 절차는 상품개발 및 기획품질 단계, 부품전개, 공정전개, 생산전개 4단계로 이루어진다. 상품개발 및 기획품질의 단계에서 CTQ(Critical to Quality)를 선정하여 CTQ에 해당된 부품전개를 한다. 부품 전개에서는 부품의 CTQ를 선정하여 공정전개 단계로, 공정전개에서는 CTP(Critical to Process)를 선정하여 생산전개 단계로, 생산 전개에서는 CTPR(Critical to Production)을 선정한다. <그림 1>은 상품개발 및 기획 품질의 단계를 보여준다. 시장의 요구(고객의 요구)로부터 얻은 데이터를 정리하고 Scene전개하여 요구품질과 품질특성을 도출한다. 도출된 각 특성은 친화도와 계통도를 거쳐 그룹핑과 새로운 아이디어를 추가하여 HOQ(House Of Quality : 품질집)를 작성하게 된다. 단계1에서 CTQ를 선정하여 단계2의 품질특성과 부품특성에 입력하고 작성하여 2단계 HOQ를 작성한다. <그림 2>는 부품전개 단계를 도식화 한 것이다. 공정전개(3단계), 생산전개(4단계)의 작성 흐름도 부품전개(2단계)와 동일하다. 물론 QFD는 수학 공식같이 고정된 형식이 아니다. 프로젝트의 특성과 필요에 따라 얼마든지 수정하여 다양하게 사용할 수 있다.



<그림 1> 상품개발 및 기획품질 (1단계)



<그림 2> 부품전개 (2단계)

2.2 QFD 전개 도구

QFD의 주요한 전개 도구는 VOC작성(Scene전개, 원시 데이터 입력), 친화도, 계통도, HOQ 등을 꼽을 수 있다.

3. 웹기반 QFD 소프트웨어 설계

본 연구에서 개발한 QFD 소프트웨어의 주요 기능은 다음 3가지로 나누어 볼 수 있다.

- (1) 고객의 소리를 수집
- (2) QFD 전개 도구
- (3) 협동작업 환경 지원 기능(다자간 응용프로그램 공유, 자료공유, 화이트보드, 다자간 채팅)

본 연구에서는 일반적으로 Stand-alone 버전의 QFD 지원 소프트웨어에서 제공하는 기능 이외에 동기적 협업을 위한 기능을 추가하여 시스템을 개발하였다. 협업은 개인이나 그룹의 유저들이 서로의 정보를 공유하고, 그것을 바탕으로 어떤 특정한 응용프로그램을 통해 업무 과정을 진행하는 것으로 주요 기능은 응용프로그램 공유가 있다[3]. 응용프로그램 공유란 엑셀이나 워드프로세서와 같이 개인용으로 만들어진 응용프로그램을 화상회의와 같은 공동 작업에 이용하는 것으로 가장 대표적인 예는 Microsoft사의 NetMeeting이 있으며 본 논문에서는 응용프로그램 공유 개념을 도입하여 시스템을 설계하였다.

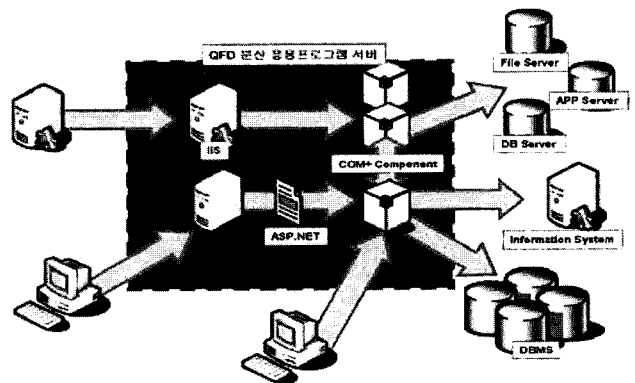
3.1 웹기반 QFD 소프트웨어의 개발환경

응용프로그램을 개발하는데 있어서 코드를 작성하는 대신 이미 만들어진 컴포넌트를 조립해서 재사용(Reusability)을 높일 수 있는 CBD(Component Based Development) 환경이 개발 패러다임으로 자리 매김하여 가고 있다[7].

이러한 개발 환경 변화에서 클라이언트/서버 구조일 경우에는 비즈니스 로직과 사용자 인터페이스를 중앙 서버에 두지 못하고 각각의 개별 컴퓨터에 설치하여야 하므로 관리 면에서 어려움이 있다. 또한 중앙통제 결여에 따라 응용프로그램의 사소한 변화가 있더라도 다시 재배포하는 불편함이 있고, 각각의 컴퓨터 별로 중요한 비즈니스 로직을 조회하는데 필요한 권한을 부여하기가 어렵기 때문에 보안이 결여 되어 있다. 그리고 모든 비즈니스 로직을 클라이언트 단독으로 처리할 경우 복잡한 로직을 실행하려면 클라이언트의 과부하가 발생한다. 이러한 문제점을 해결하기 위한 방안으로 N-Tiers 분산 환경으로 응용프로그램을 비즈니스 로직과 사용자 인터페이스를 클라이언트와 서버에 적절하게 분산함으로써 관리, 보안, 과부하 문제를 해결 할 수 있다[4].

본 논문의 분산 응용프로그램의 개발을 위해 Microsoft의 COM+ 기술을 사용하였다. COM+는 분산 컴퍼넌트를 위한 런타임 환경을 말한다. COM+의 근본적인 목적은 중요 업무 응용 프로그램과 가용성 있는 응용 프로그램을 구성하는데 사용되는 컴퍼넌트의 개발, 관리, 배포를 단순화하는데 있다. COM+는 개발자로 하여금 분산되어 있는 컴퍼넌트 기반의 응용프로그램을 신속하게 구현하고 관리할 수 있도록 여러 서비스와 관리 도구를 제공한다. 이러한 서비스들로 인해서 개발자들은 공통의 중간 계층 하부 구조를 구현하지 않고, 시스템의 업무와 응용 프로그램 로직에 집중 할 수 있다[11].

<그림3>의 QFD 분산 응용 프로그램의 구조를 나타낸 것으로 .Net(C#,VB)으로 만든 응용프로그램을 서버의 COM+ 구성요소로 등록하고 DB 서버, 응용프로그램 호스트서버 등으로 작업을 분배하도록 하며 IIS 웹서버를 통하여 웹서비스를 수행하고 ASP.Net으로 Web 응용프로그램을 개발하였다. 응용프로그램의 개발에 .Net 기술을 사용함으로써 얻을 수 있는 장점은 개발의 단순화, 배포의 단순화, 모든 부분에 XML 이용, 유니버설 데이터 액세스, 웹서비스 등의 장점을 가진다[1].

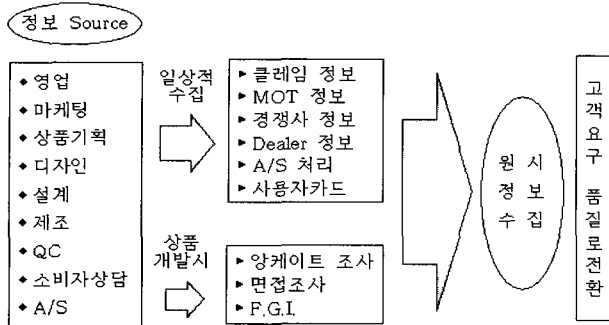


<그림 3> QFD 분산 응용프로그램 구조

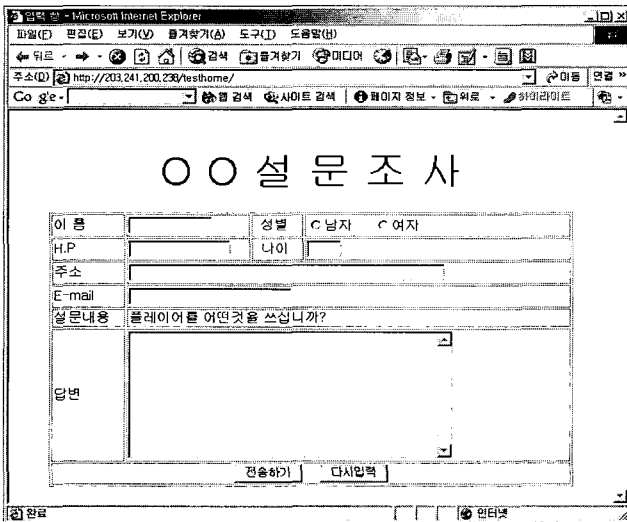
4. 웹기반 QFD 소프트웨어 구현

4.1 고객의 소리 수집기능 구현

고객 요구의 정보를 얻는 방법은 <그림 4>와 같이 다양하고 방대하다. 따라서 본 연구에서는 여러 경로를 통해 수집된 데이터를 원격지에서 입력할 수 있도록 웹상에 입력 페이지를 만들어 데이터를 수집할 수 있는 기능을 포함시켰다. <그림 5>는 e-mail을 통하여 설문 조사를 행하는 예시 화면이다.



<그림 4> 고객의 소리 수집 방법



<그림 5> e-mail을 통한 설문조사

4.2 QFD 전개 도구 구현

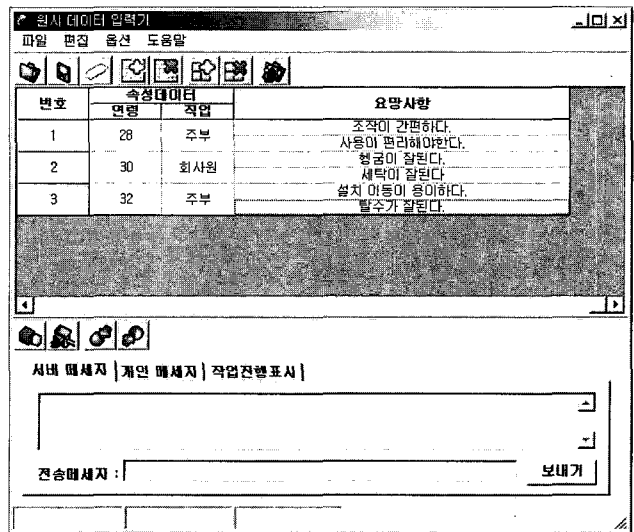
본 연구에서는 모든 프로그램(모듈)을 별개의 독립적인 프로그램 형식으로 제작하였고 서로 다른 프로그램에 의해 종속되지 않게 설계하여 프로그램의 위치가 어

디에 있어도 상관이 없도록 하였다. 또한 모든 프로그램들은 네트워크기능이 있어 필요하면 서버와의 연결 또는 다른 클라이언트들과 메시지를 주고받을 수 있으며 파일도 공유할 수 있다.

QFD 프로젝트는 고정된 형식이 아니고 다양한 전개 상황을 가지므로, 독립적인 프로그램 중에 수행하고자 하는 프로젝트에 필요한 프로그램만 조합하여 프로젝트를 진행할 수 있도록 구성하였다. 또한 새로운 지원 도구의 추가 및 기존 도구의 업데이트도 쉽게 지원되도록 설계하였다. 본 연구의 시스템은 원시 데이터 입력기, Scene 전개기, 품질특성 추출기, 진화도 작성기, 계통도 작성기, HOQ 작성기 모듈로 구성되어 있다.

(1) 원시 데이터 입력기

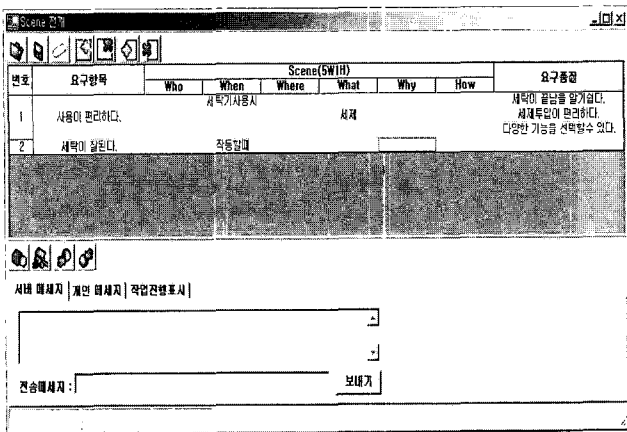
고객으로부터 수집한 정보를 입력하고 정리할 수 있는 기능을 제공한다. 판매 대리점의 고객만족도 조사, e-mail에 의해 입력된 데이터를 불러오거나 직접 입력할 수 있도록 구현하였다. <그림 6>은 본 연구의 원시 데이터 입력기 예시 화면이다.



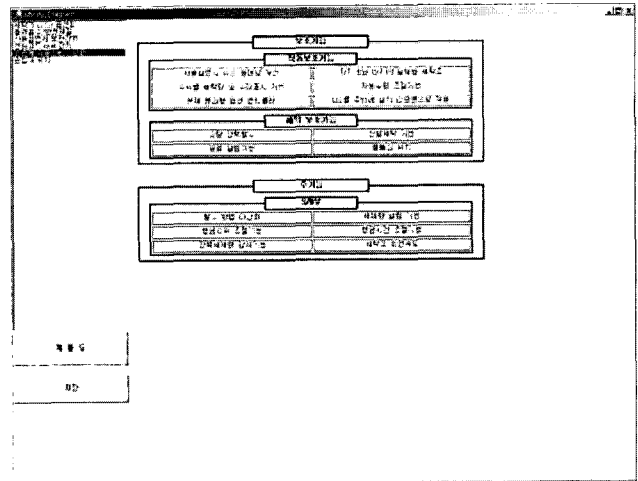
<그림 6> 원시 데이터 입력기

(2) Scene 전개기

Scene 전개는 요구품질을 추출하기 위해 사용한다. 원시데이터 입력기로부터 작성되어 저장된 파일을 불러오거나 요구항목을 직접 입력한 후 Scene전개를 하고 요구 품질을 추출한다. 다음으로 품질특성을 추출하기 위한 품질특성 전개를 위해 품질특성 추출기로 작성된 파일을 전송하거나 저장한다. 추출한 요구품질은 품질집의 요구품질로 입력될 수 있도록 구현하였다. <그림 7>은 본 연구의 Scene 전개기 예시 화면이다.



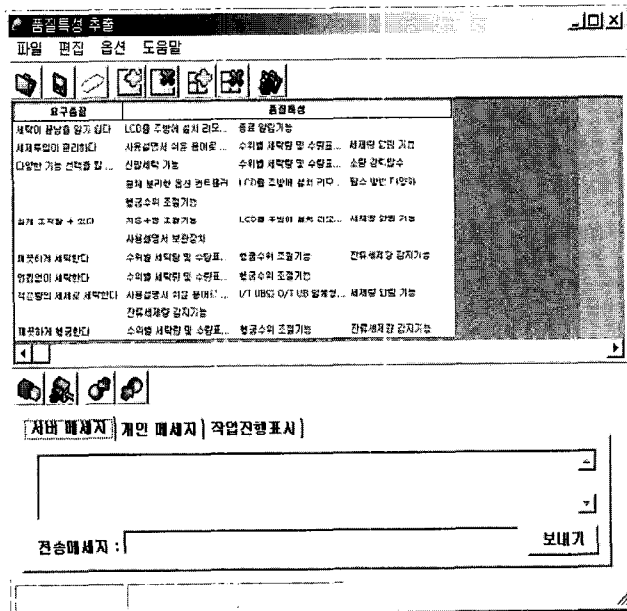
<그림 7> SCENE 전개기



<그림 9> 친화도 작성기

(3) 품질특성 추출기

품질특성 추출기는 Scene 전개로 추출한 요구품질로부터 그에 대응하는 기술적인 특성들을 추출하는데 사용한다. 먼저 요구품질을 불러오거나 입력한 후 대략 3단계 정도의 품질특성을 추출하여 작성한다. 작성된 데이터는 품질표의 품질특성으로 입력될 수 있도록 구현하였다. <그림 8>은 본 연구의 품질특성 추출기 예시 화면이다.



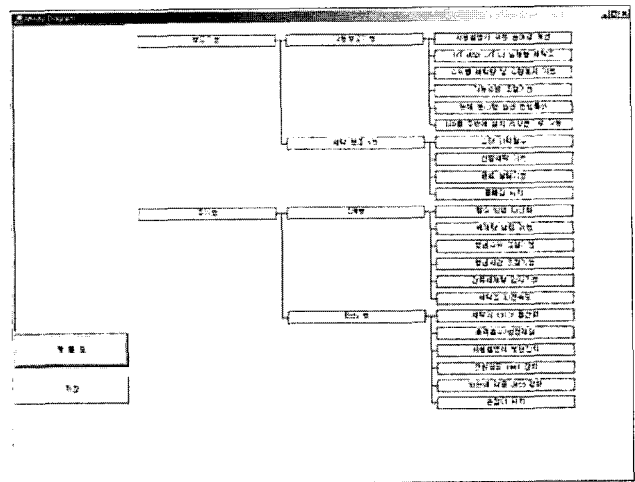
<그림 8> 품질특성 추출기

(4) 친화도 작성기

친화도 작성기는 친화도 작성을 위하여 사용된다. <그림 9>는 본 연구의 친화도 작성기 예시 화면이다.

(5) 계통도 작성기

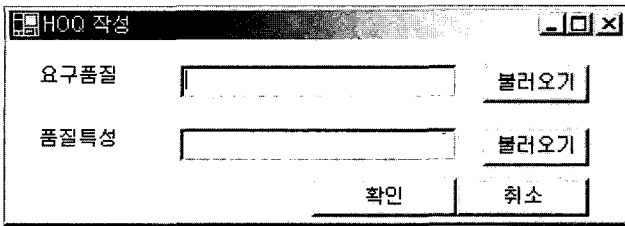
계통도 작성기는 계통도를 작성하기 위해 사용된다. <그림 10>은 본 연구의 계통도 작성기 예시 화면이다.



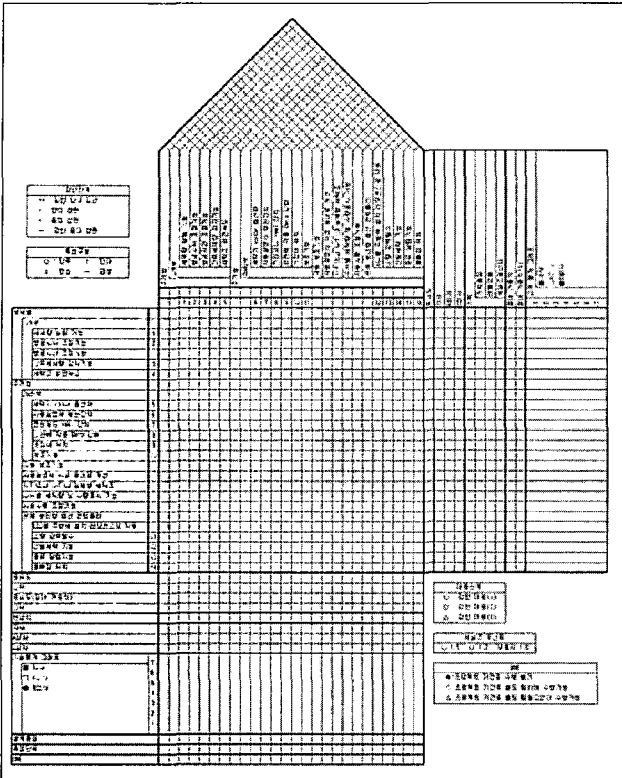
<그림 10> 계통도 작성기

(6) HOQ 작성기

HOQ 작성기는 HOQ를 작성하기 위해 사용된다. 먼저 HOQ 작성기를 실행하면 <그림 11>과 같은 대화상자가 열리고 요구품질과 품질특성은 미리 작성하여둔 파일(친화도파일, 계통도파일, 요구품질파일, 품질특성파일)을 불러온다. 확인 버튼을 클릭하게 되면 <그림 12>와 같이 HOQ를 생성하여 준다. <그림 12>의 HOQ에서 비어있는 각 cell에 정보를 입력하면서 QFD프로젝트를 진행한다. 본 연구에서 만약 친화도 파일이나 계통도 파일을 미리 작성하여 두지 않은 경우에는 수동으로 HOQ를 생성하고 만들 수 있도록 하였다.



<그림 11> HOQ 작성 대화상자



<그림 12> HOQ 작성

4.3 협업 환경 지원 기능 구현

협업 환경 구현을 위하여 QFD 지원 도구들에 네트워크 기능을 추가하여 원격지간 데이터를 주고 받을 수 있도록 개발하였고, 작업의 관리 및 동기화가 가능하도록 호스트 프로그램과 컴포넌트를 개발하였다.

호스트 프로그램과 컴포넌트의 물리적 위치는 웹서버를 담당하는 웹서버에 위치하게 되며 네트워크 기능을 가진 QFD지원 도구들은 프로젝트 팀원 각각의 PC 즉 클라이언트에 위치하게 된다. 여기서 호스트 프로그램은 서버에 위치하여 클라이언트 프로그램들이 서버를 통해 원활한 작업 수행을 할 수 있도록 적절한 작업 중계를 수행하는 프로그램을 말한다.

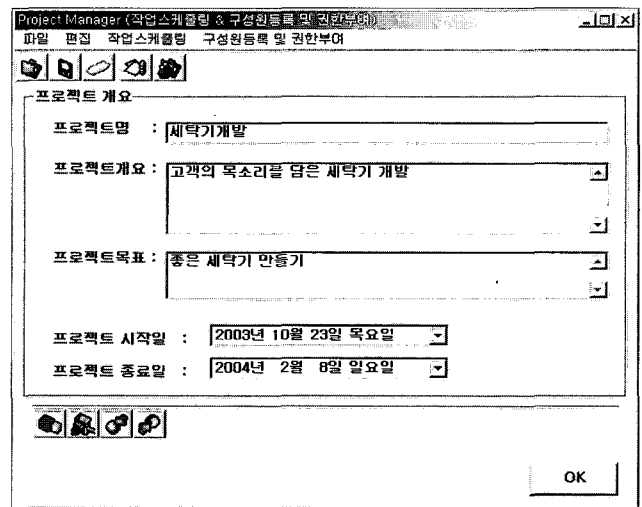
협업은 여러 사람이 한꺼번에 한 가지 일을 실시간으로 수행하는 것을 말한다. 작업을 수행 할 때 한 번에 한 사람만 작업의 내용을 수정할 수 있도록 하여야 한다. 그렇지 않으면 작업의 혼란을 가져 올수 있기 때문이다. 본 연구의 시스템은 협업을 원활하게 수행하도록 한 번에 한 클라이언트만 작업을 수정할 수 있도록 제어권을 할당하는 시스템으로 개발하였다.

4.3.1 QFD 협업을 위한 준비 및 지원기능 구현

QFD 협업 환경은 크게 협업의 준비 및 지원 기능과 협업을 수행하는 기능으로 나누어 설계하였다. 팀을 이루어 활동하는 일은 그 일과 관련하여 조직이 구성되어야 하고 일의 순서와 규칙 및 권한이 있어야 한다. 이러한 팀 프로젝트의 관리를 위해 필요한 기능을 다음과 같이 설계하였다.

(1) 프로젝트 등록

새로운 프로젝트를 진행하기 위해서는 서버에 프로젝트를 등록하여야 한다. 프로젝트는 동적으로 생성할 수 있으며 프로젝트 개요와 목표 기간 등을 설정할 수 있도록 하였다. <그림 13>은 본 연구의 프로젝트 매니저 예시 화면이다.

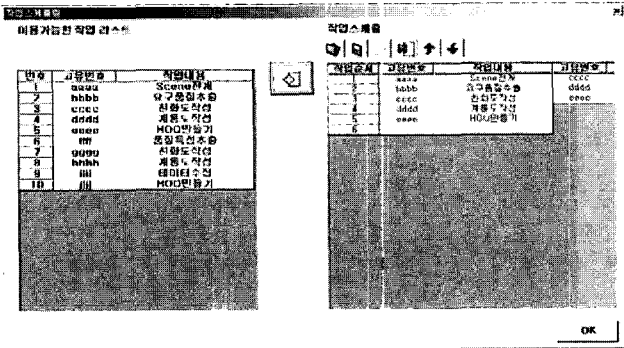


<그림 13> 프로젝트 매니저

(2) 작업스케줄링

QFD는 순차적으로 진행되는 성격이 강하다 따라서 다중 사용자 지원을 위해서는 프로젝트의 스케줄에 의해서 작업이 진행되어야 한다. 응용프로그램에서 지원하는 작업들을 미리 정의해두고 프로젝트 관리자는 프로젝트의 특성을 고려하여 작업의 앞뒤를 조합하여 스케줄링 할 수 있도록 하였다. 만약 동시에 진행되어야 하

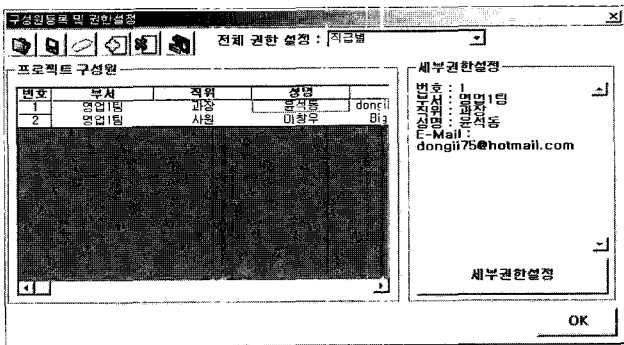
는 작업이 있을 경우를 고려하여 스케줄 입력 시 같은 열에 입력하는 구성으로 동시작업이 가능하게 하였다. <그림 14>는 본 연구의 작업스케줄링 예시 화면이다.



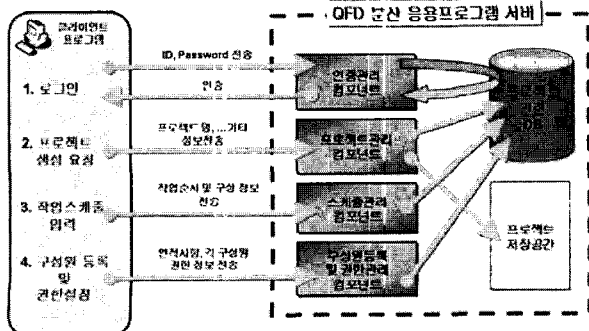
<그림 14> 작업스케줄링

(3) 구성원등록 및 권한설정

본 연구에서는 서버 쪽의 응용프로그램들은 각각 별개의 실행 파일로 구성하였고 각 응용프로그램 마다 권한을 설정 할 수 있도록 하였다. 응용프로그램 실행권한, 데이터 읽기 권한, 편집 권한 등을 각각 다르게 부여할 수 있도록 하였다. <그림 15>는 본 연구의 시스템에서의 구성원 등록 및 권한 설정 예시 화면이다.



<그림 15> 구성원 등록 및 권한설정



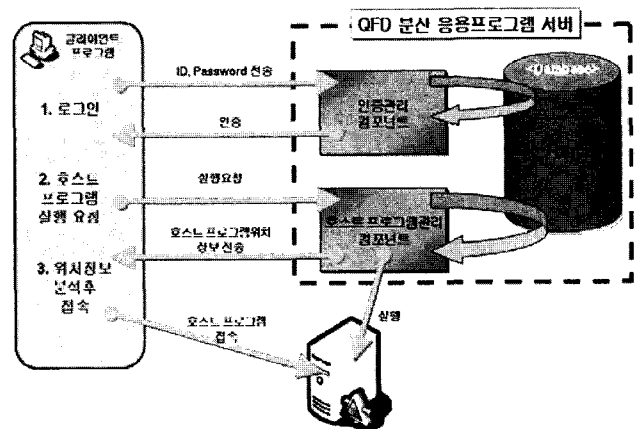
<그림 16> QFD 협업을 위한 준비 및 지원기능 설정 프로세스

<그림 16>은 본 연구의 시스템에서의 QFD 협업을 위한 준비 및 지원 기능을 설정하는 과정을 보여 주고 있다. 그 과정은 다음과 같다.

- ① 프로젝트 관리자는 응용프로그램 서버에 로그인하여 권한을 획득한다.
- ② 관리자는 프로젝트 생성을 요청한다. 이때 필요한 정보를 서버로 전송하고 서버는 정보를 받아서 프로젝트의 정보를 저장할 수 있는 데이터베이스와 프로젝트 중에 작성되는 각종 산출물을 통합 관리할 수 있는 저장 공간을 생성한다.
- ③ 관리자는 앞으로 수행할 작업내용과 스케줄을 서버에 전송하고 서버는 그 정보를 데이터베이스에 저장하고 이후 작업 수행 시 저장된 정보를 기반으로 프로젝트 일정을 수행한다.
- ④ 팀 활동을 원활히 수행하기 위해서 각 구성원의 시스템 접근 권한을 정하여 권한 정보를 서버로 전송하고 서버는 그 정보를 기반으로 각 구성원이 시스템에 접근하는 권한을 차등 적용한다.

4.3.2 QFD 협업 수행 기능 구현

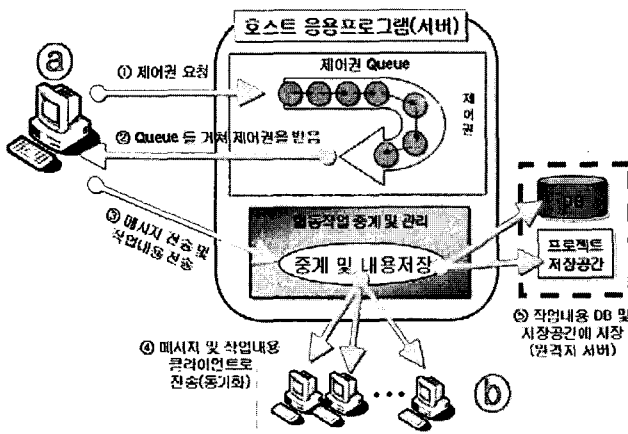
협업 수행기능은 크게 호스트 프로그램을 실행 및 종료하는 기능과 협업을 진행하는 두 가지로 나누어 설계하였다. 전술한 바와 같이 호스트 프로그램은 서버에 위치하여 실행된다. 호스트 프로그램은 협업이 진행 중일 때만 실행되어야 하며 그 실행 및 종료 권한 또한 관리가 되어야 한다. 왜냐하면 아무나 호스트를 실행하거나 종료하게 되면 서버에 부하를 주고 프로젝트 진행을 방해받을 수 있으며 보안상 문제가 발생할 수 있기 때문이다. 본 연구에서는 협업 진행을 위하여 제어권을 적절하게 주고 받으면서 작업을 실시간으로 진행해 나가도록 설계하였다.



<그림 17> 호스트 프로그램 실행 프로세스

<그림 17>은 본 연구의 시스템에서 호스트 프로그램 실행 과정을 보여 주고 있다. 그 과정은 다음과 같다.

- ① 프로젝트 관리자 또는 실행 권한이 있는 팀원이 서버에 로그인을 한다.
- ② 인증이 완료 되면 서버에 호스트 프로그램 실행을 요청 한다. 이때 어떤 호스트 프로그램을 실행할 것 인지를 전송하면 서버는 해당 호스트 프로그램의 위치정보를 디비에서 읽어 오고 그 내용을 분석하여 호스트 프로그램을 실행하고 실행정보 및 호스트 프로그램의 위치정보를 클라이언트에게 전송한다.
- ③ 클라이언트는 서버로부터 받은 위치 정보를 분석하여 호스트 프로그램에 접속한다.
- ④ 이후 다른 클라이언트들은 서버의 호스트 프로그램 관리 컴포넌트를 통하여 정보를 내려 받고 호스트 프로그램에 접속하게 된다.



<그림 18> 협업 진행 프로세스

<그림 18>은 본 연구의 시스템에서 호스트 프로그램이 실행된 후 협업이 어떤 과정에 의해서 진행되는지를 보여 주고 있다. 즉, 협업 중에서 제어권을 어떻게 획득하고 작업에 필요한 의사 교환 및 작업내용을 어떻게 주고 받는 지의 과정이다. 그 과정은 다음과 같다.

- ① 클라이언트 ①a가 호스트 응용프로그램에게 제어권을 요청하면 호스트 응용프로그램은 요청을 받아서 제어권 Queue에 요청을 넣는다. 제어권 Queue의 순서대로 제어권을 할당하고 ①a의 순서가 되면 제어권을 넘겨받는다.
- ② 클라이언트 ①a는 제어권을 획득하여 다른 모든 클라이언트들에 메시지와 작업 내용을 전달할 권한을 얻는다.

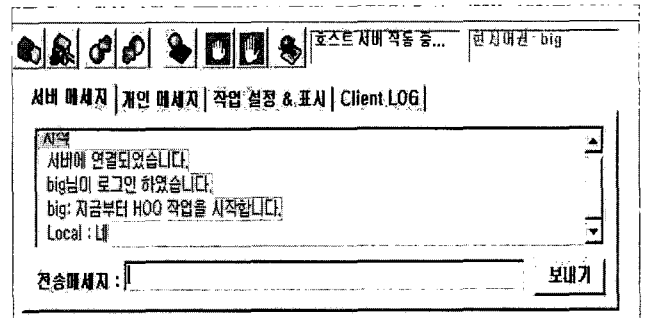
- ③ 클라이언트 ①a가 메시지와 작업 내용을 입력 하면 그 내용은 호스트 응용 프로그램으로 보내지고 호스트 응용프로그램은 받은 정보를 다른 모든 클라이언트들에 중계 한다. 이때 생성된 산출물과 과정은 모두 원격지의 DB와 지정 저장 공간에 실시간으로 저장된다.

호스트 응용프로그램은 분산된 서버중 하나에 위치하고 있으며 응용 프로그램들의 서버 역할을 한다. DB와 저장 공간이 호스트 응용프로그램 밖에 있는 것은 분산되어진 환경을 의미한다.

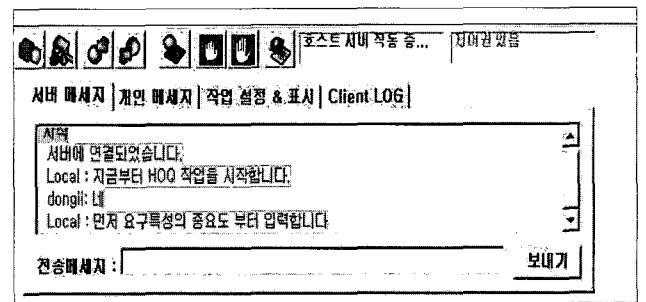
본 연구의 시스템에서 협업이 진행되는 과정을 예를 통해 살펴보기로 한다. 예에서 제어권을 가진 클라이언트를 "big"으로 표시하며 제어권을 가진 후부터의 과정을 설명하기로 한다.

(1) 의사 교환

<그림 19>는 본 연구의 시스템에서 제어권을 가진 "big"이 작업 시작을 알리고 클라이언트가 응답을 하는 예시화면이다. <그림 20>은 "big"이 QFD 작업중 HOQ의 요구특성의 중요도 작업을 입력한다고 알리고 있다. 의사 교환은 예시와 같은 채팅을 통해서 주로 이루어지고 있으며 추후 음성 채팅 화상 회의 시스템을 도입하여 보다 원활한 의사 교환이 가능하도록 할 계획이다.



<그림 19> 의사 교환 예시1



<그림 20> 의사 교환 예시2

(2) 작업 내용 동기화

제어권을 가진 작업자는 자신의 PC에서 QFD작업을 응용프로그램을 통해서 작성하기만 하면 된다. 작성된 프로그램의 내용이 실시간으로 모든 클라이언트로 전송되어 동기화가 이루어 지며 한곳에서 작업 하는 것과 같은 효과를 가지게 된다. <그림 21>은 제어권을 가진 “big”이 작업을 하는 예시 화면이고 <그림 22>는 “big”이 작업하는 내용이 실시간으로 다른 모든 클라이언트에게 동일하게 보여 지는 예시 화면이다. 이렇게 제어권을 주고받으면서 작업을 교대로 반복하여 HOQ 작성을 완성해간다. 다른 모든 QFD 작업도 스케줄에 따라서 위의 작업 순서대로 진행하여 프로젝트를 완료하게 된다.

<그림 21> HOQ작성 다중 사용자 지원 예시1

<그림 22> HOQ작성 다중 사용자 지원 예시2

5. 결 론

최근까지 QFD는 조선, 자동차, 우주 항공, 철도 등 운송 분야, 전기 및 전자 분야, 소프트웨어 개발 분야, 건설 분야, 서비스 분야, 공공 분야, 교육 분야 등등 거의 모든 분야에서 폭넓게 적용되고 있다[10]. 그리고 QFD는 팀을 이루어 노력하는 복잡한 활동이다. 그러나 상용화된 대부분의 QFD 소프트웨어는 stand-alone 방식이어서 QFD 프로젝트를 수행함에 있어서 프로젝트 팀원들이 같은 시간, 같은 장소에 있지 않으면 프로젝트를 진행하기가 어렵다. 본 논문에서는 이러한 시간과 장소의 제약에 대한 보완, 정보 활용의 용이성, 작업의 동시성을 고려하여 QFD 프로젝트를 지원 할 수 있는 방안으로 웹기반 QFD 소프트웨어를 개발하였다. 본 연구에서 개발한 시스템의 주요 특징은 다음과 같다. 첫째, 본 연구의 QFD 소프트웨어는 고객의 요구사항 수집 도구, QFD 전개도(원시 데이터 입력기, Scene 전개기, 품질특성 추출기, 친화도 작성기, 계통도 작성기, HOQ 작성기), 동기적 협업 환경 지원 도구(다자간 응용프로그램 공유, 자료공유, 화이트보드, 다자간 채팅)를 포함하고 있다. 협업 환경은 응용프로그램 공유 매커니즘을 이용하여 실시간으로 원격지간 작업을 공유하도록 하였다. 둘째, N-Tires 분산 환경의 구조로 설계하여 비즈니스 로직과 사용자 인터페이스를 클라이언트와 서버에 적절하게 분산함으로써 관리, 보안, 과부하 문제를 해결하였다. 셋째, 응용프로그램 개발은 .Net 기술을 사용하였으며, 분산 컴퍼넌트(서버측)의 개발은 COM+ 기술을 사용하여 분산 환경을 구축하여 개발 및 배포를 단순화하고 모든 부분에 XML 이용 추후 확장에도 대비하였다.

참고문헌

- [1] 김재우, 강호원, 정재선, 육창근, 오준석, Visual Basic .NET programming Bible, 영진닷컴, 2002.
- [2] 박영택, “품질기능전개의 확장에 관한 연구”, 품질경영학회지, 25(4) : 27-49, 1997.
- [3] 신영미, “분산 멀티미디어 협동 작업 환경을 위한 CORBA 기반의 보안성 있는 세션 서비스”, 석사학위논문, 포항공대, 1999.
- [4] 이명호, “N-티어 분산 환경에서 e-Commerce 설계 및 구현”, 한국산업경영시스템학회 2003년도 춘계학술대회 논문집, pp.12-17, 2003.
- [5] 이상복, 신동철, 품질기능 전개의 이론과 실무, 상조사, 2002.
- [6] 이희식, 이명주, 최용정, “비용을 고려한품질기능전개 (QFCD) 소프트웨어 개발에 관한 연구”, 한국산

- 업경영시스템학회 2005년도 춘계학술대회 논문집, pp.186-192, 2005.
- [7] 한국전산원, “분산 객체 기술 표준 연구”, 1996.
- [8] Herzwurm, G., Reiß, S., Schockert, S., “The Support of Quality Function Development by the Customer Oriented Evaluation of Software Tools”, Transactions from the Fifteenth Symposium on Quality Function Deployment and the Ninth International Symposium on QFD '03, Orlando, Florida, USA, pp.139-158, 2003. ([http : //www.qfd-id.de/en/articles/evaluation_tools/qfd_software_english.pdf](http://www.qfd-id.de/en/articles/evaluation_tools/qfd_software_english.pdf))
- [9] Huang, G.Q., MAK, K.L., “Synchronous Quality Function Deployment(QFD) over World Wide Web”, Computers and Industrial Engineering, 42(2-4) : 425-431, 2002.
- [10] Chan, L.K., Wu, M.L., “Quality Function Deployment : A Literature Review”, European Journal of Operational Research, 143 : 463-497, 2002.
- [11] Matthew Bortniker, James Conard, Professional Visual Basic .NET Transactions, 정보문화사, 2002.