

☒ 응용논문

정보시스템 개발자를 위한 GUI 생성에 대한 연구

- A Study on GUI Generation for Information System Developer -

민 병 도*

Min, Byeong Do

이 재 봉**

Lee, Jae Bong

이 종 석***

Lee, Jong Suk

Abstract

GUI(Graphic User Interface) have an influence on end-user environments in information system. Consistency is one of the most important GUI characteristics. Consistency should apply across the different media which form the total user interface. User can get a lot of benefits through consistency. To achieve consistency of the total interface, the interface standards are very important.

The goals of this study is to suggest GUI's development guidelines through analysis of real fields using ERP package in a electric assembly line. The guidelines fall into four functions: inquiry, registration, modification, deletion. The guidelines can be used for better system development later and as verification tool in GUI development.

1. 서론

정보시스템에서 데이터의 조회, 수정, 입력 그리고 삭제를 사용자가 직접 행할 시 GUI(Graphic User Interface) 환경이 매우 중요하게 된다. 기존 GUI를 설계 및 개발함에 있어 처음부터 완성된 시스템을 개발하지 않고, 시제품을 작성하여 사용자의 평가를 받는다. 기존 소프트웨어의 전형적인 모델에 있어서 개발의 대부분의 시간을 다양한 중간 작업제품의 재결정을 위해 소비하므로 이에 대한 대책이 필요하다. 더구나 사용자 인터페이스에 대한 적절한 지침들을(guidelines) 마련해 놓지 못한다면 개발자 자신의 경험에 입각한 인터페이스를 제작함에 따라 사용자에게 오류를 발생시키고, 시스템의 일관성에 반대적인 영향을 끼친다. 이러한 문제점을 해결하기 위하여 일관된 지침을 이용할 필요가 있다.

정보시스템을 개발하는데 있어 두 가지 측면을 고려하여야 한다. 우선 전체 시스템의 일부

* 두원공대 공업경영과

** 대우 정보시스템

*** 두원공대 산업디자인과

본 기능적인 사항에 초점을 두거나, 전체적인 사용자 인터페이스 측면에서 개발할 수도 있다. 올바른 시스템 개발을 위해서는 기능적인 면이나 사용자 인터페이스 측면에서 동시에 고려해야 하는 것이 올바른 것이라 할 수 있다. 본 연구에는 두 단계에 걸쳐 GUI의 개발지침을 연구하였다. 우선 기능적인 측면은 기존 현장의 분석을 통하여 이루어 졌고, 본 연구에서는 전체 사용자 인터페이스 측면에서 일관성을 갖도록 하였다.

본 연구의 대상범위는 도구(power builder, Visual C++등)를 사용하여 GUI를 구성하는데 필요한 윈도우를 개발할 경우를 대상으로 한다. 또한 연구대상이 되는 객체들은 실제 개발현장(전자조립 라인)에서 사용되었던 대표적인 객체를 중심대상으로 하였고 이에 추가적인 일반 객체들을 추가로 고려하였다. 일반적인 표준객체와 GUI를 개발을 위해 사용되는 대표적인 객체는 다음과 같다.

GUI를 개발을 위해 사용되는 대표적인 객체(괄호로 묶은 것들은 같은 부류로 취급함)

: Command button(Picture button), Check box, Radio button, Static text, Picture group box (line, Oval, Rectangle), Single Line Edit(Edit Mask, Multiline edit), Rich text edit, Horizontal scroll bar(Vertical scroll bar), Drop down list box(Drop down data window), List box, Tree view, tab, data window, Graph, OLE, User 객체중 yymmdd_term_scroll

2. GUI 설계를 위한 기본 지침 및 기존 현장의 분석

2.1 GUI 설계의 기본 지침

GUI의 설계에 있어 중요한 점은 사용자를 중심으로 해야 한다는 것이다.[6] 이러한 관점을 직접적인 조정(direct manipulation)이라 한다. GUI를 개발하기 위해서는 여러 종류의 지침을 참조할 수 있다. 이러한 지침들은 일반적인 지침(general guidelines), 개발시스템의 종류에 따른 범주특성별 지침(category-specific guidelines), 개인시스템에 대한 개별특성별 지침(product-specific guidelines)으로 구분될 수 있다.[7] GUI를 개발하기 위하여 GOMS(goals, operators, methods, selection rules)라는 분석적인 방법이 있다.[8] 참고문헌 [9]에서는 GOMS가 실제계에서 매우 유용하다는 것을 언급하였다. 또한 GUI를 설계할 때는 사용성(Usability) 측면에서 중요하게 생각해야 할 사항은 다음과 같이 요약될 수 있다.[7]

- 학습성(Learnability) - 시스템은 사용자가 쉽게 배워 각 업무를 적용할 수 있다.
- 효율성(Efficiency) - 시스템은 효율적이어야 한다.(업무의 생산성 향상)
- 기억성(Memorability)- 시스템은 기억하기가 용이하여야 한다.
(잠시 시스템을 사용하지 않더라도 다시 사용할 경우에 쉽게 적응할 수 있어야 한다.)
- 에러(Errors) - 에러발생이 적어야 한다.
- 만족도(Satisfaction) - 사용자가 만족하여야 한다.

Mac OS8의 H. I.(Human Interface) Guidelines[4]는 윈도우 상에서 사용되는 객체들의 배치 문제에 대하여 언급되어있다. 문헌 [4]에서는 "Reader's eye tends to move from the upper-left corner of the dialog box to the lower right" 에 따라 윈도우상의 전반적인 구조는 좌측상단에는 중요부분이 되는 객체를 배치하고, 우측하부에는 해당 기능을 수행하기 위한 명령버튼이나 데이터 윈도우등의 출력이나 동작을 명령하는 요소를 배치해야 한다고 언급하고 있다. GUI를 디자인 측면에서 설계하기 위해서는 다음과 같은 기준이 필요하다.[10]

- 가시성의 원칙 - 사용자에게 어떤 기능이 가능하고, 어떻게 조작되어야 하고, 조작한 상태가 어떠한지에 대한 정보의 투명성을 의미한다.

- 단순의 원칙 - 사용상의 조작구조를 요구되는 노력을 최소화하여야 한다.
- 예견의 원칙 - natural mapping이라고 하는데, 즉, 사용자가 실제로 조작을 하기 이전에 조작한 결과를 미리 예견할 수 있어야 한다.
- 일관성의 원칙 - 무리한 표준화가 아니라, 차별화 요건을 그대로 받아들여 일관성을 주는 것이며, 제품통합디자인(PI : Product Identification)에 중요한 요소가 된다.
- 제한점(constraints) 이용의 원칙 - 조작에러 유형의 제한사항을 두어 여타 가능한 선택의 여지를 줄여 조작방법을 명확히 하는 것이다.
- 지식분배의 원칙 - 요구되는 정보특성에 따라 정보의 일부분은 사용자로 하여금 학습하도록 하고, 나머지 부분은 제품상에 나타나도록 유형의 지식을 안배하는 것이다.
- 조작오류(errors)의 원칙 - 발생된 오류의 원인을 쉽게 이해하도록 이를 쉽게 고칠 수 있도록 하고 그 원인을 최소화시키도록 설계한다.

2.2 현장 분석

본 연구의 분석대상으로는 전자조립라인에 설치된 ERP 시스템이다. 이 시스템은 기술정보, 생산계획, 구매관리, 자재관리의 4가지 모듈로 구성되어있다. 기존의 큰 문제점은 GUI에 필요한 윈도우를 구현할 시에 각 모듈 개발자간의 취향, 경험의 차이와 사용자의 요구에 따라 윈도우들의 형태가 상이하다는 것이다. 즉, 다수의 개발자가 동시에 정보시스템에 사용되는 GUI를 구현할 때, 개발자 개인적인 특성 및 각 사용자의 의견을 수렴하다보니 사용 윈도우의 형태가 차이가 난다는 것이다. 비록 기본적인 모듈의 사용으로 기본 윈도우의 큰 형태는 동일하나 윈도우상에 사용되는 객체와 이들의 배열은 상이하야, 같은 업무를 처리하는 곳에서도 다양한 데이터 처리를 해야 하는 어려운 점이 발생되었다. 따라서, 이러한 문제점을 해결한 방안으로서 개발자간의 미리 일정한 지침 또는 표준안의 필요성이 대두되었다.

전자조립라인에 설치된 ERP 시스템의 각 모듈에 사용된 윈도우들을 상호 비교하여 분석한 결과, 다음 아래와 같은 대표적인 차이점 및 애로사항을 파악하였다.

- 전반적인 윈도우 형태에서 조회 기능을 행하는 방법이나 순서는 동일하나, 이를 위해 사용되는 객체들(자식 윈도우, 탭페이지, 기타 윈도우등)의 상이성
- 마우스의 오른쪽 버튼의 사용유무. 즉, 해당 기능메뉴의 pop up여부
- 공통적으로 사용된 객체들의 기능이 불일치
- 메시지 박스의 활용정도 및 메시지 내용의 불일치
- 데이터 윈도우, 리스트박스, 라인 에디터 등 입력 및 편집되는 각 형태의 데이터 및 사용방법에 상이점 발생(디폴트값 유무의 차이등)
- 각 윈도우를 사용하는 관리자, 작업자의 환경에 따라 세부 처리내용이 결정되므로 전체적인 측면의 통합화에 어려움이 발생

앞에서 언급한 문제점을 해결하기 위하여 다음 두 가지를 중점사항으로 삼았다. 첫째, 사용자는 조작방법을 쉽게 배우고 사용하기 편해야 하며, 어느 정도의 심미적인 요소도 갖추고 있어야 할 것이다. 둘째, 개발자의 관점에서는 개발시간 및 노력이 적게 소요되고, 추후 보전작업이 용이하게 이루어질 수 있도록 단순화, 모듈화가 이루어져야 한다. 이러한 중점사항을 조화시켜 기존 ERP 시스템에서 발생되었던 GUI의 문제점을 해결하고, 사용성이 뛰어난 시스템을 개발할 수 있을 것이다.

3. 본론

정보시스템에서 GUI를 구현할 시, 사용자의 관례가 존재하지 않는 경우가 발생된다. 이러한 경우, 사용자가 친숙한 입출력 양식을 이용해야 한다. 사람들은 대상을 인식한다. 그러나 다른 사람들과 그 대상을 공유하기 위해서 그 대상에 관한 공통적인 견해가 생성될 수 있도록 그 대상이 무엇인가에 관한 상호합의가 있어야 한다. 동일한 정보시스템에서 사용되는 GUI의 일관성이 중요하게 되는 것도 바로 이 시점이다. 왜냐하면 동일한 사양은 사물들이 같은 방법으로 인지되고 서술적일 것을 요구하기 때문이다. 공통적인 견해가 부족한 곳에서는 표준화가 불가능하기 때문이다. 따라서 실제 현장에서 사용되는 윈도우의 기본 기능을 살펴보고 이에 따른 적절한 윈도우 형태를 결정하여야 한다. 그러나 기존의 개발절차에서는 이러한 점이 바르게 적용되지 않아서 사용자 및 개발자에게 많은 애로점과 혼란을 가져왔다.

GUI를 구현할 때에 앞에서 제시한 개발 지침을[7][10] 기반으로 다음 두 가지의 방법을 고려할 수 있다.

첫째, 윈도우를 설계할 때에 참조할 수 있는 설계지침을 만들어 객체들이 일관성을 갖도록 배열될 수 있도록 한다.

둘째, 공통도들의 사용자 객체의 기능을 보장하여 차이점을 없애는 것이다.

전자는 큰 범위에서 문제를 해결하는 방안이며 후자는 큰 범위에서의 문제점이 해결된 후 이를 보장해주는 역할을 한다.

본 연구에서는 정보시스템 개발 할 때, GUI(Graphic User Interface)를 구현하기 위해 사용되는 윈도우의 올바른 개발을 위한 지침을 제안하고자 한다. 제안된 개발 지침은 실제 국내 전자조립라인 현장에서 사용되고 ERP 시스템의 분석을 통하여 이루어졌다. 본 연구는 범주특성별 지침(category-specific guidelines)을 제안하는 것이다.

본 연구의 현장분석 대상이 되는 전자조립라인에 개발·적용된 ERP 시스템에서 사용자가 직접 수행하는 대표적인 4가지 기능은 조회, 추가, 수정, 삭제이다. 각 기능의 대략적인 수행비율을 현장 개발자들의 의견 및 작성된 프로그램 분석을 통해 파악한 결과, 조회 60%, 추가(등록) 15%, 수정 20%, 삭제가 약 5%를 차지한다. 추가 및 수정의 경우는 조회가 이루어진 상태에서 자식 윈도우를 열어 수행하며, 삭제의 경우는 확인작업을 팝업 윈도우를 사용토록 되어있다. 따라서 가장 빈도가 많고 중요도가 높은 조회 기능을 수행할 윈도우의 설계 및 구현이 가장 중요하다. 추가, 수정 및 삭제 기능은 대부분이 조회가 이루어진 상태에서 발생하므로 조회 기능을 수행할 윈도우를 고려하여 결정하여야 한다. 따라서 본 연구에서는 조회 기능을 우선 중점사항으로 삼아 추가·수정·삭제에 대한 윈도우 형태를 상호 연관시켜 결정한다.

기존의 윈도우 작성 표준안의 내용을 살펴보면 간략하게나마 GUI의 표준안을 마련하고자 하는 노력이 보인다. 이들 내용은 폰트, 색상, 데이터, 윈도우의 border, report에 대해 간략하게 언급하고 있다. 하지만 규정된 내용 외에 명시되어야 할 사항은 많다. 예를 들면 윈도우상에 사용되는 객체들의 색상에 대한 언급은 없으며, 객체들의 배치에 관한 언급은 없다.

본 연구에서는 일반 상용소프트웨어 및 기존 ERP 시스템에서 사용되었던 윈도우들을 분석하여 다음과 같은 공통적인 형태를 우선 결정하였다..

- 윈도우에 배치되는 모든 요소들에 대하여 각각의 독립성을 유지하되 자형, 색상, 크기는 가독성과 완성도를 고려하여 굴림체, 검은 색, 10, bold를 사용한다.
(그룹박스는 굴림체, 10, normal)
- 입력을 받는 객체(예를 들면 SLE¹⁾, DDLB²⁾, DDDW³⁾)의 색상은 흰색으로 하며,

출력을 하는 객체는 반드시 갈색계통을 사용한다.

- 그룹 박스를 사용 시 해당 제목을 반드시 입력한다.
- 윈도우상의 기본구조는 좌측상단에는 중요부분이 되는 객체를 배치하고, 우측하부에는 해당 기능을 수행하기 위한 명령버튼이나 데이터 윈도우등의 출력이나 동작을 명령하는 요소를 배치한다.[4]
- 조회작업을 수행하는 메인 윈도우(main window), 자식 윈도우(child window), 반응 윈도우(response window)상에 도움말 버튼을 좌측에 배치하고 명령버튼을 우측으로 배치하여 해당 윈도우가 수행하는 기능에 대한 설명이나 도움말에 직접 접근할 수 있도록 사용자의 편의를 도모하고 시각적인 가치를 향상시킨다. 윈도우의 개발 지침(Guidelines)은 각 기능별로(조회, 추가, 수정, 삭제) 구분하였다.

3.1 조회 작업

조회작업을 수행하는 윈도우들은 입력이나 선택을 받는 객체, 조회된 결과를 보여주는 데이터 윈도우와 객체, 해당 데이터 윈도우에 종속되는 요소나 상세 내역에 대한 객체들로 구성되어 있다. 따라서 좌측상단에는 조회의 시발점이 되는 객체를 배치하고, 우측하부에는 해당 기능을 수행하기 위한 명령버튼이나 데이터 윈도우등의 출력이나 동작을 명령하는 요소를 구분하여 배치한다.[4] 따라서 좌측 상단 부분부터 입력이나 선택을 받는 객체들을 배치하고 중단 부분에 조회 결과가 보여지는 데이터 윈도우와 관련된 객체, 하단 부분은 해당 데이터 윈도우에 종속되는 요소나 상세 내역에 관한 객체들을 배치한다. 그리고 조회작업에서 SLE와 같이 사용된 명령버튼을 제외하고 해당 윈도우상에 버튼을 배치하지 않는 것을 원칙으로 한다.(마우스의 오른쪽 버튼의 팝업 메뉴사용) 또한 조회에 사용되는 입력 요소들의 배치 우선 순위는 중요도가 되는 부분, 범위를 크게 나눌 수 있는 부분, 객체의 크기가 작은 것을 좌측상단부터 배치한다.

조회작업에서 사용되는 윈도우를 위한 지침은 다음과 같다.

- 윈도우의 제목은 적절한가?
- 입력 또는 선택을 받는 부분이 상단부분에 배치되어 있는가?
 - 해당기능을 수행하기 위한 가장 적절한 객체인가?
 - 사용된 색상, 서체 그리고 크기는 원칙을 따랐는가?
 - 핵심이 되는 요소를 우선적으로 배치했는가?
 - 객체길이가 상대적으로 긴 것은 우측에 배치되어 있는가?
 - 적절한 그룹화가 되었으며 그룹박스를 사용한 경우 제목을 주었는가?
- 출력을 하는 부분이 중단부분, 하단부분에 배치되어 있는가?
 - 사용된 색상, 서체와 크기는 원칙을 따랐는가?
 - 자유형태의 데이터 윈도우를 사용했을 경우 좌우 및 상하 배치가 균등한가?
 - 하나의 윈도우로 표현하기 힘든 경우 선택한 대안이 최적인가?
(탭 조정, 버튼, 트리뷰, 팝업메뉴 중에 적절한 선택 : <부록 1> 참조)
- 상세 내역이나 기타 종속되는 요소들이 있을 경우 하단에 배치되어 있는가?

1) SLE(Single Line Editor) 하나의 라인으로서 문자를 받는 객체
 2) DDLB(Drop Down List Box) : SLE의 변형된 형태로서 SLE의 오른쪽에 화살표가 붙어져 있어 여러 가지 항목중 하나를 선택하여 입력받는 형태의 객체임
 3) DDDW(Drop Down Data Window) ; DDLB와 유사한 형태로서 DDLB는 보여주는 항목이 하나이지만 이것은 몇 개의 항목들을 보여줄 수 있음.

- 사용된 색상은 원칙을 따랐는가?
- 중단부분에 출력되는 요소들과의 연동관계는 적절한가?
- 팝업메뉴를 사용할 때에 마우스의 오른쪽 버튼기능이 부가되었는가?
- 메시지 박스에서 조회진행 과정을 기술해주고 있나?

다음 그림 1은 조회작업의 샘플화면으로서 상단부분에 선택을 하는 부분이 배치되어 있으며, 좌측에 트리뷰(treeview), 그리고 선택한 조건에 타당한 결과를 보여주는 탭페이지(tab page)부분이 있다. 상단부분의 배치를 살펴보면 조회의 일반적인 선택기준이 되는 내용을 라디오(radio) 버튼으로 구성하여 좌측상단에 배치되었고, 이에 종속적으로 두 번째 기준이 되는 생산기간을 입력받는 객체는 우측상단에 배치되어 있다. 좌측중단부분에는 상단 부분에서 선택한 조건에 타당한 결과가 조회된다. 트리뷰에 조회된 결과 중에 하나를 선택하면 이에 따른 결과 값이 바로 탭 페이지에 조회된다. 또한 관련 있는 데이터 내용은 그룹으로 구성하여 전체적인 내용을 쉽게 이해하도록 하였다. 즉, 조회작업 윈도우를 이용할 시, 사용에러를 없애고 전체적인 내용을 자연스럽게 볼 수 있도록 하기 위하여 사용자의 시선이 좌측 상단부분에서 우측하단으로 이동하도록 윈도우 화면을 구성하였다.

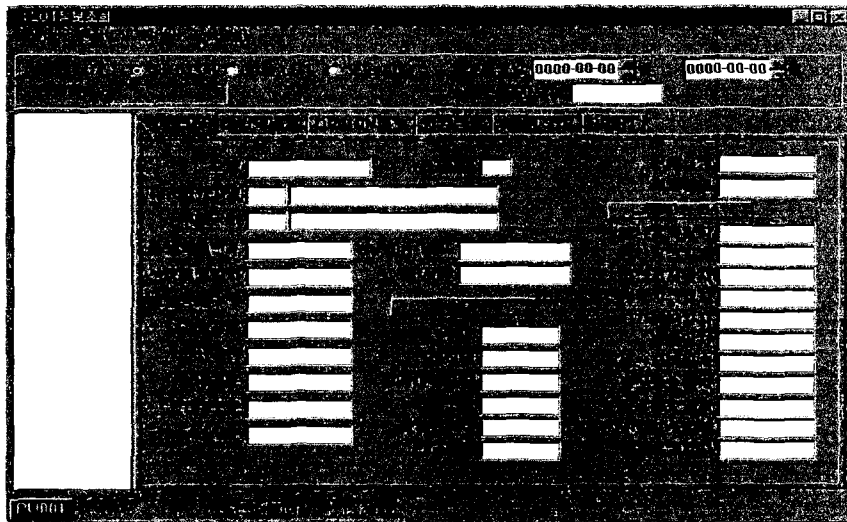


그림 1. 조회작업의 샘플화면

3.2 추가 및 수정작업

추가 및 수정작업은 데이터의 변경이 일어나므로 사용자의 입력에러를 최대한 방지하여야 한다. 따라서 추가 및 수정 작업을 동일한 윈도우에 같이 병행하는 것보다 독립적인 윈도우로 구성한다.

추가 입력작업은 해당 데이터항목을 입력받을 수 있는 자식 윈도우를 개별적으로 열어서 데이터 입력을 정확하게 할 수 있도록 반영을 하였다. 또한 여러 번의 데이터의 입력을 위하여 우측하단에 추가 및 종료버튼을 두었다. 수정작업은 자식윈도우를 열어 메인 윈도우의 데이터를 가져와서 수정작업을 거친 후 이를 다시 메인 윈도우에 반영하는 형태를 취한다. 또한 동일한 데이터에 대하여 여러 번의 수정을 할 수 있도록 처리 및 종료의 버튼을 구성하였다. 또한 사용자의 편의를 위해 하단 부분에 도움버튼을 좌측에 배치하였다.

추가 및 수정작업을 위한 윈도우에서 사용되는 객체의 전반적인 배치는 조회화면에서 언급한 경우와 동일하고, 그 이외의 추가적인 개발 지침은 다음과 같다.

- 윈도우상에 사용된 요소들의 배치는 조회작업에서 설정한 원칙을 지키고 있는가?
- 윈도우 제목은 적절한가?
- 반영과 취소의 기능버튼이 마련되어 있는가?

그림 2는 추가작업의 샘플화면으로서 상단부분에 조회를 위한 입력부분이 있어 기존의 내용을 확인할 수 있다. 입력된 LOT번호에 대한 내용이 그룹박스에서 조회되고 하단부분에는 추가하고자 하는 내역을 입력하는 부분이 있다. 즉, 입력을 받는 부분은 바탕을 흰색으로 하고 결과를 보여주는 부분은 회색색상을 사용하여 명확하게 구분하여 사용자의 오류를 방지하였다. 전반적인 배치는 조회의 경우와 동일하고 도움말 버튼과 추가 및 종료 버튼을 하단부분에 배치하였다.

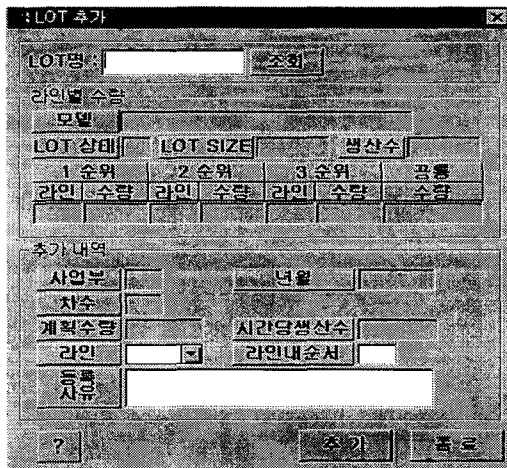


그림 2. 추가작업의 샘플화면

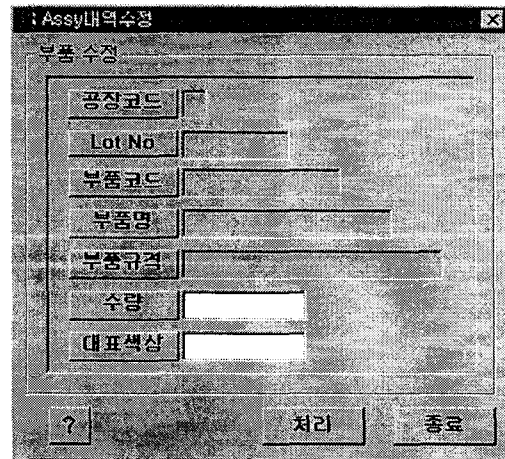


그림 3. 수정작업의 샘플화면

그림 3에서 보듯이 수정작업은 추가작업과 유사한 화면 구조를 가지고 있으므로 추가작업과 마찬가지로 조회되는 부분과 입력을 요구하는 부분을 분명하게 한다.

데이터의 추가 및 수정작업에 있어서 데이터 입력의 오류가 발생할 시에 이를 사용자에게 알리기 위한 반응 윈도우(Note Alert 박스)를 이용한다. 조회 데이터에 대하여 불필요한 행동을 할 경우, 더 이상 진행될 수 없음을 알리기 위하여 다른 반응 윈도우(Stop Alert 박스)를 이용하여 사용자에게 알려준다.

3.3 삭제작업 및 반응 윈도우(response window)

대부분의 삭제작업은 데이터 윈도우에서 내역을 조회한 상태에서 해당항목의 삭제를 선택한다. 이때, 사용자에게 삭제작업을 확인할 수 있는 피드백(feedback) 메커니즘을 제공하기 위하여 반응 윈도우를 사용한다. 사용자의 삭제를 확인 시, 사용하는 반응 윈도우의 형태는 제목 부분에 삭제작업의 종류를 명시하고, 윈도우의 내부에 삭제하고자 하는 항목과 내용을 기술하되 밑줄에 삭제의 복구여부를 나타낸다. 하단부분에는 2 개의 명령버튼(YES/NO)을 위치시키고, 여기에 심미적인 요소와 사용자의 편의성을 고려하여 하단좌측에 아이콘을 배치하고 2 개

의 버튼을 하단우측에 위치시킨다. 삭제작업의 경우 추가 및 수정을 위해 개발된 모듈을 이용해서 사용하며, 배치나 색상은 3.2에서 언급할 형식으로 따른다. 특히, 삭제작업에 사용되는 반응 윈도우는 나타내는 정보는 축약적인 의미보다는 서술적인 언어로 정확하게 전달할 수 있어야 한다.

반응 윈도우는 사용자가 데이터의 조회, 수정, 추가 그리고 삭제작업을 할 때, 사용자의 오류를 방지하기 위하여 사용되는 윈도우이다. 반응 윈도우의 내용은 사용자에게 정확한 의미를 전달해야 하므로 축약적인 것보다는 서술적으로 정확하게 표현해야 한다. 또한, 사용자가 반응 윈도우의 발생이유를 정확하게 인지할 수 없는 경우, 도움말 기능(help)을 통하여 부연설명을 할 수 있도록 하여야 한다.

분석 결과에 따르면 반응 윈도우는 크게 3가지의 박스로 분류될 수 있다.

- Caution alert 박스 : 삭제작업에서 사용되는 박스로서 데이터의 손실이 발생할 수 있는 상황에서 주로 사용된다. Yes와 No, help의 3가지 버튼을 갖추고 있으며 삭제할 대상을 설명해주는 부분은 붉은 색(red color), bold체를 사용하고, (예: **KW82E0001 LOT를 삭제하시겠습니까?**) 하단에 복구될 수 있음과 없음은 검은 색(black color), normal체로 표시한다.(예: LOT 삭제 후 복구될 수 없음)
- Note alert 박스 : 데이터 손실의 위험은 없으나 사용자에게 유용한 정보를 제공할 경우 사용될 수 있다. 이 박스의 서체는 caution alert 박스와 같으나 검은 색(black color)을 사용하며, 버튼은 help, OK를 배치하는데 데이터가 없거나 사용자의 입력 오류일 경우 사용한다.
- Stop alert 박스 : Action이 더 이상 진행될 수 없을 경우 사용되며 글씨는 붉은 색(red color), bold체를 사용하여 한 줄로 표시한다. (예: **LOT BOM이 생성된 상태이므로 삭제될 수 없음**) 사용되는 버튼은 help와 OK 버튼을 사용한다.

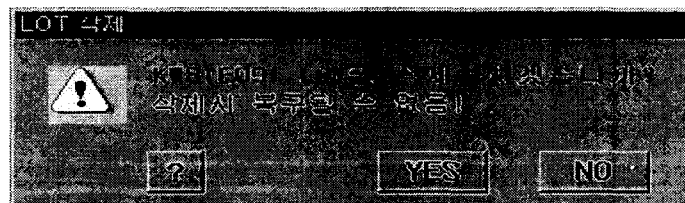


그림 4. 삭제작업의 sample 화면

삭제에서 반응 윈도우를 사용하는 경우, 삭제대상을 분명히 기술하고 사용자의 주의를 충분히 환기시킬 수 있도록 색상과 배치를 해야 한다. 추가 및 수정작업에서 사용되는 반응 윈도우도 같은 구성을 가지도록 하여 일관성을 유지하도록 하고, 사용된 아이콘만으로도 충분히 의미를 전달할 수 있어야 한다.

4. 결론

정보시스템 개발의 규모가 크고, 많은 수의 개발자가 참여할 경우에는 개발자들간의 활발한 의사전달을 통해 개발 모듈간의 일관성이 유지될 수 있도록 해야 한다. 이러한 작업들은 본격적인 개발에 앞서 공통모듈이나 표준안의 작성 등을 통해 미리 행해져야하는 선행과정이다. 이 목적을 충족시키기 위하여 실제로 개발되어 사용하고 있는 ERP 시스템의 분석을 통하여 GUI

설계를 위한 지침(guidelines)을 제안했다.

본 연구에서 제시한 지침들은 대상 시스템의 개별적인 특성을 고려하였기 때문에 일반적인 시스템에 적용하기는 다소 무리가 있다. 그러나 전자조립라인을 위한 시스템 개발에 있어서 다수의 개발자들이 GUI를 구현하기 위해 사용되는 윈도우들간의 일관성을 가질 수 있도록 하였다. 또한 추후에 개발되는 전자조립라인을 위한 ERP 시스템에서 모듈이 다를 경우라도 사용자 인터페이스를 일관성 있게 구성함으로써 사용자의 오류 및 능숙하게 사용될 수 있을 때까지 소요되는 시간을 줄일 수 있을 것이라 판단된다.

본 연구에서 제시된 지침(Guidelines)을 개발자가 올바르게 적용하기 위해서는 일정한 형식을 가진 양식으로 존재하여야 한다. 현재 이러한 양식의 결정을 위하여 추가작업이 이루어지고 있다.

본 연구의 궁극적인 목적은 GUI를 위한 윈도우 프로그램을 생성함에 있어서 자동생성에 이르기까지 자동화를 이룰 수 있는 기반을 마련하고 다수의 개발자가 쉽게 이해할 수 있는 기준을 마련하여 프로그램의 유지보수 작업을 용이하게 하고자 하는 것이다. 이를 위하여, 첫째, 각 지침 항목들의 상호관련성에 대한 분석, 둘째, 각 지침 항목들을 체계적으로 적용할 수 있는 방법론의 추가적인 연구가 필요하다.

5. 참고 문헌

[1] Marcus Aron : Graphic Design for Electronic Documents and User Interfaces, ACM Press, Addison-Wesley, NewYork(1992)
 [2] Hinsai Mark, The secret of Effective GUI Design, 지산사, 1995
 [3] 다윈시스템 개발관련(ERP 개발관련 자체보고서)
 [4] Mac OS8 Human Interface Guildelines, www.apple.com
 [5] Graphic Design, 장동훈, www.nexon.co.kr/Chang/
 [6] Sheiderman, B. (1983). "Direct manipulation: A step beyond programming languages. IEEE Computer 16, 8(August),57-69.
 [7] Nielsen, J., Usability Engineering, Academic Press, Inc., 1993, pp. 91~92
 [8] Card, S. K., (1983). The Psychology of Human-Computer Interaction", Lawrence Erlbaum Associates, Hillsdale, NJ.
 [9] Gray, W. D., John, B. E., and Atwood, M. E.(1992), "an overview of a validation of GOMS. Proc. ACM CHI'92. Conf.(Monterey, CA, 3-7 May), 307-312
 [10] 이건표 , "A Conceptual Framework on User-Interface Design", 디자인연구, No. 2,1994

< 부록 1> Control type의 장단점

Control Type	장 점	단 점
Tab Control	모든 선택들을 보여줄 수 있음. 텍스트에 추가하여 아이콘을 보여줄 수 있음.	수평적 공간을 많이 차지함. 확장성에 한계가 있고 작은 아이콘만 사용 가능함.
버튼	모든 선택을 동시에 보여줄 수 있음. 아이콘과 텍스트를 보여줄 수 있고 tab controls보다 공간을 적게 차지함	수평·수직 공간을 많이 차지하고 확장성에 한계를 가짐
Tree Views	확장성이 무제한이고 간단한 아이콘과 텍스트 사용가능.	수평·수직 공간을 많이 차지하고 모든 선택을 보여주진 못함
Popup 메뉴	모든 선택을 동시에 보여주며 확장성이 무제한이며 공간을 작게 차지함	사용자가 모든 선택을 보기를 원할 경우에 사용되고 작은 아이콘과 텍스트로만 보여질 수 있다.