



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사 학위논문

웹 콘텐츠 접근성에 기반한
전자결재시스템의
설계 및 구현



2009년 8월

부 경 대 학 교 산 업 대 학 원

전 산 정 보 학 과

서 미 영

공학석사 학위논문

웹 콘텐츠 접근성에 기반한
전자결재시스템의
설계 및 구현

지도교수 윤 성 대

이 論文을 工學碩士 學位論文으로 提出함

2009년 8월

부 경 대 학 교 산 업 대 학 원

전 산 정 보 학 과

서 미 영

이 논문을 서미영의 공학석사
학위논문으로 인준함

2009년 8월 26일



주 심 이학 박사 박 홍 복 (인)

위 원 이학 박사 이 경 현 (인)

위 원 이학 박사 윤 성 대 (인)

<차 례>

표 차례	iii
그림 차례	iv
Abstract	vi
I. 서 론	1
1.1 연구의 배경 및 목적	1
1.2 연구의 방법	4
II. 관련연구	5
2.1 웹 접근성 개요	5
2.1.1 웹 표준과 웹 접근성	5
2.1.2 웹 접근성의 표준 지침	8
2.1.3 국내·외 웹 접근성 표준화 현황	12
2.1.4 국내 장치법의 단계적 범위	14
2.2. 전자결재시스템 개요	16
2.2.1 전자결재시스템의 개념	16
2.2.2 전자결재시스템의 구축 목적	17
III. 전자결재시스템의 설계	19
3.1 전자결재시스템의 설계 방향	19
3.1.1 설계 시 고려사항	20
3.1.2 업무 처리 과정	20

3.2 전자결재시스템의 설계 구조	22
3.2.1 메뉴 구성	22
3.2.2 화면 레이아웃 설계	23
3.2.3 데이터베이스 설계	24
IV. 전자결재시스템의 구현	27
4.1 개발 환경	27
4.2 디자인 구현	29
4.2.1 문서의 표준 정의	29
4.2.2 영역 범위 및 이동 경로	31
4.2.3 웹 표준화 적용	32
4.3 프로그램 구현	43
4.4 기존 시스템과의 차이점	45
4.5 구현 결과화면	46
V. 결론 및 향후 연구	50
참고문헌	52

<표 차례>

표 1. 국내 장차법의 단계적 범위	15
표 2. 전자결재시스템의 정의	17
표 3. 전자결재시스템의 주요 메뉴	22
표 4. 하드웨어 구성 환경	28
표 5. 소프트웨어 구성 환경	28
표 6. 기존 전자결재시스템과의 차이점	45

〈그림 차례〉

그림 1. 국내 웹 접근성 표준화 현황	13
그림 2. 전자결재시스템 구축 효과	18
그림 3. 전자결재 업무 흐름도	21
그림 4. 전자결재시스템의 기본 화면 레이아웃	24
그림 5. 논리적 데이터베이스 모델링	25
그림 6. 문서등록 테이블	26
그림 7. 전자결재시스템 하드웨어 구성도	27
그림 8. 표준화 문서의 기본 정의	30
그림 9. 콘텐츠의 시각적 이동 경로	31
그림 10. 계층형 문서 구조	32
그림 11. 전자결재시스템의 기본 레이아웃	33
그림 12. 반복 메뉴 건너뛰기	34
그림 13. 전자결재시스템의 문서 기안 소스	36
그림 14. 전자결재시스템의 문서 기안 페이지	38

그림 15. 스크립트 비교	39
그림 16. 일반적인 HTML 형태의 코딩 방식	40
그림 17. 웹 표준을 적용한 프로그램 코딩 방식	41
그림 18. 전자결재시스템의 결재 프로세스	43
그림 19. 인터넷 익스플로러 v6.0	46
그림 20. 인터넷 익스플로러 v7.0	47
그림 21. 모질라 파이어폭스	47
그림 22. 오페라	48
그림 23. 구글 크롬	48
그림 24. 애플 사파리	49
그림 25. 넷스케이프 내비게이터	49

**Design and Implementation of Electronic Approval System
Based on Web Contents Accessibility**

Mi-Young Seo

**Department of Computer and Information
Graduate School of Industry
Pukyong National University**

Abstract

As information technologies and the internet have come into common use, the majority of the population including the disabled, preschoolers, and senior citizens gain daily access to the Internet to find information on the web. Many of the existing web sites have been developed focusing solely on their functional and aesthetic aspects, without sufficiently taking into account user interface features for those who need special requirements. For example, people with sensory and other disabilities can have a difficulty in using such systems.

We standardize an electronic approval system, one of the main management information systems, according to the Web Contents Accessibility Guidelines published by W3C(World Wide Web Consortium) and by the Korean IABF(Information and Telecommunications Accessibility Promotion Standard Forum) in this paper. For those who cannot use a mouse, the system is modified to be operated only via a keyboard, and the facility of page zoom in/out is added for people with low vision and elderly people. In addition, the overall accessibility to the system is improved by providing the pages with the same view across heterogeneous browser environments such as Internet Explorer and FireFox.

The size of file is reduced through the modularization and minimization of tags, which in turn leads to faster page loading and saving of the server space. It is demonstrated that the web sites developed following the standard guidelines help people find information on the web more quickly and easily, and thus make web contents more available to all users, including disabled users.



I. 서론

1.1 연구의 배경 및 목적

전자상거래 및 인터넷 뱅킹, 온라인 교육, 전자정부의 민원서류 발급 등 IT기술의 발전과 더불어 일상생활에서 필요한 대부분의 정보들은 인터넷을 통해 처리가 가능해졌고, 간편하게 습득할 수 있게 되었다.

현재 인터넷이용률을 살펴보면 만 6세 이상 전 국민의 77.1%, 장애인(210만 명)의 51.8%가 각종 콘텐츠 등을 얻기 위해 인터넷을 활용하고 있으며, 2008년 12월 말 기준으로 전자조달 나라장터를 통한 전자입찰이 약 30만 건, 전자민원발급이 560만 건이며, 금융기관에 등록된 인터넷뱅킹 고객 수는 5,260만 명으로 2007년 4,470만 명에 비해 16% 증가(한국은행) 하였다. 또한 만 12세 이상 인터넷 이용자 중 60.6%가 인터넷을 통해 상품이나 서비스를 구매(한국인터넷진흥원)한 경험이 있었다[12]. 일반 사용자들은 이런 다양한 정보들을 네트워크가 연결된 컴퓨터만 있으면 손쉽게 이용 가능하지만 장애인(2,104,889명 : 보건복지가족부 2007년 12월) 및 노령자(481만363명 : 2007년 말 65세 이상 전체인구대비 9.9%)의 경우[10]에는 기술적이거나 신체적인 제약(시각, 청각, 지체, 언어장애 등)으로 인해 정보 획득에 어려움이 따르고 있다.

현대 사회는 정보화, 고속화됨에 따라 경제 활동에 참여하는 수가 점차 늘어나고 있는 추세이며, 사람들의 인식변화 및 법제화에 따라 장애인에 대해서도 기업에서의 고용 형태가 더 활발해 질 것으로 예상되고 있지만 현재의 웹사이트 구조나 웹 기반의 업무 프로그램들을 살펴보면 아직 체계

화, 전문화되어 있지 않아 다수의 장애우가 이용하기에는 많은 불편함이 따른다.

미국(1990), 호주(1992), 영국(1995), 홍콩(1995), 스웨덴(1999), 노르웨이(2001) 등의 국가에서는 이미 장애우 차별 금지법이 제정되어 일상 및 사회생활 전반에서 발생되고 있는 차별에 대해 표준화, 인식제고, 인력양성, 법제도 개선 등의 대책이 마련되어 있으나 국내의 경우는 법제나 구제의 미흡으로 인해 장애우가 보호받지 못하고 있고 웹의 물리적 접근성에 대한 사회적 배려도 부족하다. 정보화 사회에서 웹 콘텐츠에 대한 접근 가능 여부는 정보격차 뿐 아니라 교육격차로 인한 빈부격차, 계층 간 갈등 심화 등 여러 가지 심각한 사회적 문제를 낳을 수 있다[13]. 이에 국내에서는 2005년부터 웹 접근성의 기술표준을 마련하여 표준화 작업을 진행하고 있지만 중앙행정기관, 광역 지자체를 제외한 기초 지자체 및 전자정부 사이트, 민간 사이트들에 대한 접근성은 아직까지 미비한 실정이다[14].

장애인 차별금지 및 권리구제 등에 관한 법률(이하 장차법)이 2008년 4월 11일부터 의무적으로 시행됨에 따라 모든 인터넷 사용자는 동등하게 사이트에 접근할 수 있어야 하며 어떠한 차별도 받지 않도록 보장되어야 할 것이다. 이는 웹 사이트 운영자 및 디자이너, 프로그래머 등 모두에게 점차 중요한 개념으로 인식되고 있는 부분이다[15].

국내 웹 표준의 또 다른 문제 요인을 살펴보면 웹사이트에서 제공하는 문서들이 대부분 비표준으로 제공되기 때문에 브라우저 및 운영체제에 따라 호환성 문제가 발생하는 것을 볼 수 있다. 1990년 대 중반 소위 웹브라우저 전쟁기간 동안 인터넷 익스플로러(IE: Internet Explorer)와 넷스케이프(NN: Netscape Navigator)간의 비표준을 기반으로한 경쟁 이후 시장이 IE 독점 상태가 되면서 IE 전용 기술을 사용하여 개발된 사이트들이 많이 등장하게 되었고 이 사이트들은 다른 브라우저에서는 웹 문서를 제대로 표

시할 수 없는 문제를 야기시켰다[16]. 2009년 4월 기준 웹브라우저의 이용 현황을 살펴보면 전 세계적으로 IE의 경우는 66.10%, Firefox 22.48%, Safari 8.21%, Chrome 1.42%, Netscape 0.82%, Opera 0.68%, 기타 0.29%의 비율을 보이고 있는데 이는 2008년 8월 IE 72.15%, Firefox 19.73%, Safari 6.37%, Chrome 0%, Netscape 0.72%, Opera 0.74%, 기타 0.29%[9]에 비해 IE의 이용률은 점차 낮아지고 타 브라우저들의 이용률은 높아지는 양상을 보이고 있는 것을 확인할 수 있었다. 아직까지 국내에서는 마이크로소프트사의 제품 의존도가 높아 IE를 많이 이용하고 있는 실정이지만 점차 사용자의 시스템 환경 및 요구사항이 다양해짐에 따라 웹 문서가 다른 조건에서 동일한 결과를 나타내기 위해서는 웹사이트의 표준화 작업은 반드시 필요한 부분으로 대두되고 있다.

본 논문에서는 전자결재의 문서제공 특성 상 시력1급 장애(전맹)를 제외한 시각 및 지체 장애우, 노령자, 일반 사용자들의 웹 사이트 접근성을 고려하여 W3C 및 국내에서 제정한 웹 표준을 기반으로 기업의 대표적인 업무 프로그램 중 하나인 전자결재시스템을 설계하고 구축하는 과정을 연구해 보고자 한다. 본 연구를 통해 사이트를 표준화시킴으로써 장애우를 비롯한 많은 사용자에게 기업 내·외부의 업무 처리 시 편리한 작업환경을 제공할 것으로 기대한다.

1.2 연구의 방법

기존의 비(非) 표준화로 구축된 전자결재시스템은 브라우저의 형태에 따라 서로 다른 결과를 보여주었고, 장애우의 경우에는 접근이 쉽지 않는 등 여러 가지 문제점이 발생하였다. 장애우 및 일반사용자가 전자결재시스템을 이용할 경우 접근성을 높이고 동일한 결과를 보여줄 수 있는 방안으로 웹 표준 지침을 준수하여 사이트를 설계하고 구축하는 것이다.

본 논문의 목적을 달성하기 위해 수행하고자 하는 연구 내용을 요약하면 다음과 같다.

첫째, 웹 접근성을 높이기 위해 W3C 및 국내에서 제정한 표준화 내용을 탐구한다.

둘째, 웹 접근성에 따른 표준 기술 파악 및 전자결재시스템에 대한 개념을 이해한다.

셋째, 웹 표준을 적용하기 위한 전자결재시스템의 요구사항 및 화면 레이아웃 등 설계의 기본 방향을 정한다.

넷째, 웹 페이지를 읽어 들이고 그 페이지를 해석할 수 있도록 디자인 및 프로그램을 설계한다.

다섯째, 웹 표준을 적용하여 전자결재시스템을 구축한다.

여섯째, 기존의 전자결재시스템과의 차이점을 분석하고 본 논문의 결론과 향후 과제를 기술한다.

II. 관련연구

본 장에서는 기업에서의 대표적인 업무 프로그램 중 하나인 전자결재시스템에 대해 살펴보고 사용자들의 웹 접근성을 높이기 위한 방안으로 국내·외에서 지정한 표준 지침 및 현황, 장차법의 단계적 적용범위에 대해 살펴본다.

2.1 웹 접근성 개요

전 세계적으로 웹 사이트는 헤아릴 수 없을 정도로 많이 존재하고 있으며 매일 새롭게 만들어지고 있다. 이런 사이트들은 개발자 혹은 의뢰 고객에 따라 서로 다른 환경에서 구축되어지며, 개발이 완료된 사이트의 경우에는 사용자의 컴퓨터 환경(운영체제 및 브라우저 등)에 따라 다르게 보여진다. 여기서 다르게 표현된다는 것은 프로그램의 기능적 요소보다 화면에 보여지는 디자인적 요소 및 사용 방식에 따른 것으로 이는 사이트를 표준화시킴으로써 해결될 수 있다. 사이트를 표준화시킨다는 것은 웹 접근성을 높인다는 의미이다.

2.1.1 웹 표준과 웹 접근성

(1) 웹 표준

웹 표준의 사전적 의미로는 월드 와이드 웹(World Wide Web)의 측면을 서술하고 정의하는 공식 표준이나 다른 기술 규격을 가리키는 일반적인 용어으로써 웹 디자인 및 개발과 관계가 있다. 수많은 상호 의존성이 있는 표

준과 규격들 가운데 일부는 단지 월드 와이드 웹으로만 끝나는 것이 아니라, 인터넷의 관리 측면이기도 하며 이러한 표준들은 직·간접적으로 웹 사이트, 웹 서비스 개발과 관리에 영향을 주고 있다. 이러한 것들 모두 "웹 표준"이라고 부르지만 웹 표준으로 이동하는 것을 찬성하는 사람들은 사용성과 접근성에 직접 영향을 미치는 더 높은 수준의 표준에 초점을 두는 경향이 있다.

웹 표준은 W3C(World Wide Web Consortium)[1] 단체에서 규정한 웹 기술 사양에 대한 규칙으로 웹 브라우저 종류 및 버전에 따른 기능 차이에 대해 호환이 가능하도록 제시되었다. W3C와 여러 웹 표준 단체들은 웹 기반의 콘텐츠를 해석하고 생성하기 위해 구조적 언어(XHTML: eXtensible Hypertext Markup Language), 표현적 언어(CSS: Cascading Style Sheet), 객체 모델(DOM: Document Object Model), 스크립트 언어(ECMAScript: European Computer Manufacturers Association Script), 부가적인 표현 언어(MathML: Mathematical Markup Language, SVG: Scalable Vector Graphics)에 따른 기술[2-5]을 제공한다.

사이트를 웹 표준 형식으로 만들면 CSS와 HTML 소스의 분리로 코드가 간략해지고 크기가 줄어들어 빠른 페이지 로딩이 가능해진다. 그만큼 서버의 저장 공간도 적게 차지되고, 회선 트래픽도 절약할 수 있어 느린 회선을 사용하는 사용자들에게 높은 접근성을 준다. 또한 스타일시트를 변화시켜 PDA, 프린트, 웹브라우저 등 서비스 형태에 따라 다양한 화면을 제공할 수 있다. 모듈화 된 코드로 인해 유지보수가 쉬워지며 비용도 감소될 뿐만 아니라 소스가 표준화되었기 때문에 검색엔진의 접근성이 향상되어 그에 따른 순위도 상승되는 장점이 있다.

각각의 다른 컴퓨터 환경에서 동일한 결과를 출력하기 위해서 사이트는 표준화되어야 한다. 한 예로 usb 메모리는 제조업체마다 외형은 달라도 포

트가 국제 표준이므로 전 세계 어디서라도 사용할 수 있다. 즉, 표준은 어떤 형태를 구성하는데 있어 최소한의 기준이고 약속이다.

(2) 웹 접근성

현재 대부분의 웹 사이트와 웹 어플리케이션은 신체적·기술적인 장애 및 기기종의 컴퓨터 환경 등으로 인해 접근성을 저해하는 요소들을 갖고 있으며, 이로 인해 많은 장애인들은 웹에 접근하지 못하고 있다. 또한 인터넷을 이용하는 모든 사용자들이 경제적 여건 등의 이유로 최신 사양의 고성능 컴퓨터를 사용하는 것은 아니기 때문에 그에 따른 배려도 필요한 실정이다. 저 사양 컴퓨터를 사용하는 사람들도 큰 불편 없이 정보를 접하고 서비스를 받을 수 있도록 하는 것과 마우스를 민첩하게 다루기 어려운 노령자의 이용에 불편이 없도록 해드리는 것도 접근성의 문제이다.

웹 접근성이란 장애인, 고령자 등 어떠한 사용자라도 웹 사이트에서 제공하는 정보에 일반인과 동등하게 접근하고 이용할 수 있도록 하는 웹 사이트 접근 환경 및 수준을 의미한다. 여기서 장애인 신체적 장애는 물론, 언어나 키보드, 마우스 등을 사용할 수 없는 장애, 또한 웹 환경의 종류나 버전이 다르므로써 콘텐츠를 이해하는데 어려움이 발생하는 경우도 포함된다[17].

웹 접근성을 준수한다는 것은 시각장애우의 경우 사이트 내 이미지 등 시각 정보에 대한 설명들을 대체 텍스트로 제공하여 의미 전달이 가능하게 하며, 청각 장애우는 동영상에 캡션(자막)을 제공하여 음성정보에 대한 내용을 확인할 수 있게 한다. 마우스를 사용하지 못하는 지체 장애우의 경우에는 키보드 이용을 보장하고 저 시력자나 노령자의 경우에는 글자의 확대/축소 기능을 제공하여 사이트의 접근을 원활하게 한다. 또한 일반 사용자와 콘텐츠 제공자에게는 구조화되고 의미 있는 리소스를 제공하여 다양한

편의성과 이익을 안겨준다[13].

웹 접근성을 향상시키기 위해서는 사이트에서 제공되는 모든 정보에 대해 장애우가 차별 없이 이용할 수 있도록 웹 접근성에 대한 인식 제고와 수준을 높이기 위한 전문교육이 필요하며, 고객 신뢰도 및 만족도, 사회적 공헌과 국내 웹 개발 환경 개선 및 관련 기술 발전을 촉진하기 위하여 품질마크를 부여하도록 해야 한다.

2.1.2 웹 접근성의 표준 지침

W3C는 국제 사실 표준화기구로 웹 관련 표준화를 담당하고 있으며, 웹의 진화를 위한 표준, 각종 규격과 참고용 소프트웨어 생산을 통한 WWW 제품들 간의 상호 운용성 증진을 추구하는 산업계의 표준을 제정한다[17].

그 일환으로 W3C에서는 1997년 장애우 등의 접근성 제고를 위해 웹 접근성 이니셔티브(WAI : Web Accessibility Initiative)를 설립하여, 1999년 5월 웹 콘텐츠 접근 가이드라인(WCAG : Web Content Accessibility Guidelines) 1.0[6]을 제정하였으나 현재의 웹 기술을 수용할 수 없는 부분도 있고, 장애우가 사이트를 이용할 경우 사용성과 만족감을 보장하지 못한다는 지적이 계속적으로 제기된데 따라 2008년 12월 WCAG 2.0[7]을 새로이 제정하기에 이르렀다. WCAG 2.0은 기존 WCAG 1.0 기준과 달리 장애우의 실제적인 사용성에 초점을 맞추었고 인터넷 기술의 발전 속도에 맞는 신기술 수용 부분까지 고려되었다. 현재 국내에서는 WCAG 1.0의 초안을 기준으로 하고, 부분적으로는 미국 장애인 재활법 508조를 참조한 웹 콘텐츠 접근성 지침 항목 1.0을 표준으로 채택하고 있다.

영국, 호주, 유럽연합(EU)에서는 국제표준화기구인 W3C의 웹 접근성 이니셔티브의 웹 콘텐츠 접근성 지침(WCAG 1.0과 WCAG 2.0)을 표준으로

채택·시행하고 있으며, 미국이나 일본의 경우는 WCAG 1.0이나 2.0을 기초로 자국의 실정에 맞게 변형하거나 추가하는 형태의 웹 접근성 표준을 마련하여 시행하고 있다[18].

국내에서는 2002년 1월 정보통신부 고시로 제정된 '장애인·노인 등의 정보통신 접근성 향상을 위한 권장지침'과 2005년 12월 민간 표준화 포럼인 정보통신 접근성 향상 표준화 포럼(IABF: Information and Telecommunications Accessibility Promotion Standard Forum)[11]에서 '한국형 웹 콘텐츠 접근성 지침 1.0'을 제정하여 웹 사이트에 접근하려는 장애우가 일반인과 동등하게 웹 콘텐츠에 접근하여 자유롭게 이용할 수 있도록 웹 사이트 설계자 및 운영자, 콘텐츠 개발자에게 국가 표준에 따른 기술 구현 방법 등을 쉽게 이해하고 준수하도록 도움을 주고 있다.

국내에서 제정한 웹 콘텐츠 접근성 지침 항목[11]은 다음과 같다.

(1) 인식의 용이성

웹사이트에서 서비스하고 있는 모든 콘텐츠는 누구나 쉽게 인식할 수 있도록 설계되어야 한다.

① 텍스트 아닌 콘텐츠(non-text contents)의 인식

텍스트 아닌 콘텐츠 중에서 글로 표현될 수 있는 모든 콘텐츠는 해당 콘텐츠가 가지는 의미나 기능을 동일하게 갖추고 있는 텍스트로도 표시되어야 한다.

② 영상매체의 인식

시간에 따라 변화하는 영상매체는 해당 콘텐츠와 동기 되는 대체 매체를 제공해야 한다.

③ 색상에 무관한 인식

콘텐츠가 제공하는 모든 정보는 색상을 배제하더라도 인지할 수 있도록 구성되어야 한다.

(2) 운용의 용이성

웹 콘텐츠에 포함된 모든 요소들의 기능은 누구나 쉽게 사용할 수 있어야 한다.

① 이미지 맵 기법 사용 제한

이미지 맵 기법이 필요할 경우에는 클라이언트 측 이미지 맵을 사용하며 서버 측 이미지 맵을 사용할 경우에는 동일한 기능을 하는 텍스트로 구성된 대체 콘텐츠를 제공해야 한다.

② 프레임의 사용 제한

콘텐츠를 구성하는 프레임의 수는 최소한으로 하며, 프레임을 사용할 경우에는 프레임별로 제목을 붙여야 한다.

③ 깜박거리는 객체 사용 제한

콘텐츠는 스크린의 깜박거림을 피할 수 있도록 구성되어야 한다.

④ 키보드로만 운용 가능

키보드(또는 키보드 인터페이스)만으로도 웹 콘텐츠가 제공하는 모든 기능을 수행할 수 있어야 한다.

⑤ 반복 내비게이션 링크(repetitive navigation link)

웹 콘텐츠는 반복적인 내비게이션 링크를 뛰어넘어 페이지의 핵심부분으로 직접 이동할 수 있도록 구성하여야 한다.

⑥ 반응시간의 조절기능

실시간 이벤트나 제한된 시간에 수행하여야 하는 활동 등은 사용자가 시간에 구애받지 않고 읽거나, 상호작용을 하거나 응답할 수 있어야 한다.

(3) 이해의 용이성

사용자들이 가능한 한 쉽게 이해할 수 있도록 콘텐츠나 제어 방식을 구성해야 한다.

① 데이터 테이블 구성

데이터 테이블은 테이블을 구성하는 데이터 셀의 내용에 대한 정보가 충분히 전달될 수 있어야 한다.

② 논리적 구성

콘텐츠의 모양이나 배치는 논리적으로 이해하기 쉽게 구성하여야 한다.

③ 온라인 서식 구성

온라인 서식을 포함하는 콘텐츠는 서식 작성에 필요한 정보, 서식 구성 요소, 필요한 기능, 작성 후 제출 과정 등 서식과 관련한 모든 정보를 제공해야 한다.

(4) 기술적 진보성

구성한 콘텐츠는 웹 브라우저의 종류, 버전 등에 관계없이 사용될 수 있어야 한다.

① 신기술의 사용

스크립트, 애플릿 또는 플러그 인(plugin) 등과 같은 프로그래밍 요소들은 현재의 보조기술의 수준에서 이들 프로그래밍 요소들의 내용을 사용자에게 전달해줄 수 있을 경우에만 사용하여야 한다.

② 별도 웹사이트 제공

콘텐츠가 항목 (1)①에서 (4)①에 이르는 13개 검사 항목을 만족하도록 최대한 노력하였으나 해결되지 않는 부분이 남아있다면 텍스트만의 콘텐츠를 제공하는 웹 페이지(또는 웹사이트)를 별도로 제공해야 한다.

2.1.3 국내·외 웹 접근성 표준화 현황

미국은 웹 접근성 제고를 위해 다양한 법제도를 마련해 왔다. “장애인 재활법 508조”[8]는 웹 접근성을 직접적으로 언급한 법으로 연방부처나 기구가 전자 및 정보기술을 개발, 조달, 유지, 사용할 때는 지나친 부담이 되지 않는 한 사용하는 기술의 종류에 상관없이 장애인 직원 또는 장애를 가진 민원인이 접근하고 활용할 수 있어야 한다고 명시되어 있다. §1194.22는 웹 기반 인트라넷과 인터넷 정보 및 응용 프로그램에 대한 기준을 마련하고 있으며 16개의 지침으로 구성되어 있다.

영국은 1995년 제정된 장애인 차별금지법을 통해 고용, 상품, 시설, 서비스제공, 교육, 교통수단 등에 대한 장애인 차별금지를 포괄적으로 규정하고 있다. 특히 19조에서는 서비스제공자가 일반인에게 제공되는 서비스를 장애인에게 제공을 거절하는 것, 장애인에게 제공되는 서비스의 수준이나 방식을 일반인과 달리하는 것 등을 불법으로 간주하고 있으며, 1999년 영국 전자정부를 추진하는 e-envoy에서 영국 정부의 웹 사이트 접근성 준수를 위한 가이드라인을 제정·공표하였다.

호주는 1992년 장애인 차별금지법을 제정하여 고용, 교육, 공공서비스 등의 모든 영역에서 장애인에 대한 직간접적인 차별을 금하고 있다. 특히 웹 접근성을 위해서 인권 및 동등기회 보장위원회가 국제 표준화기구인 W3C의 가이드라인을 표준으로 채택할 것을 권고하고 있다.

유럽연합은 1999년에 발표한 e-europe 계획의 10대 추진과제 디지털 환경에 대한 장애인들의 성공적인 참여를 실현한다는 계획을 수립했다. 특히 장애인의 웹 접근성에 대한 관심을 가져 정보사회에서 달성하고자 하는 목표를 모든 사람을 위한 정보화 사회로 설정하고 이를 달성하기 위해 장애인을 위해 웹사이트가 준수해야한다는 사항을 포함하고 있다.

일본은 1999년 정보통신 액세스빌리티(Accessibility) 패널에서 정보장벽을 제거하기 위해 WCAG 일부규정이 포함되어 있는 액세스빌리티 가이드라인을 발표했다. 그 후 2002년 정보 및 통신네트워크 사회설립을 위한 기본법이 제정되었고, IT기본법으로 불리는 이 법률 8조에서 누구나 이용할 수 있는 웹 사이트의 기본조건을 구체적으로 제시하고 있다. 특히 2004년 5월 일본규격협회의 정보기술표준화센터에서는 정보기술 분야의 접근성향상을 위한 공동지침, 정보처리장치 접근성향상을 위한 지침 및 웹 콘텐츠의 접근성 향상을 위한 지침(JIS X8341-3 : 2004) 등 3가지 분야의 규격을 일본공업규격으로 제정하였다[19].

국내에서의 웹 접근성 표준화[20]는 웹 접근성을 고려한 콘텐츠(인터넷 웹 콘텐츠 접근성 지침 WCAG 2.0, 2005년), 웹 접근성을 고려한 웹 저작도구(한국형 웹 저작도구 접근성 지침 K-ATAG: Korean Authoring Tool Accessibility Guidelines) 1.0, 사용자 에이전트(브라우저, 미디어 플레이어, 장애인 정보통신 보조기기 등)로 크게 나뉘어 발전되고 있으며, 웹 접근성 기반 조성을 위해 표준, 교육, 실태조사, 품질마크 등에 중점을 두고 있다.



(그림 1) 국내 웹 접근성 표준화 현황

웹 접근성을 고려한 콘텐츠에서는 모든 사람들이 어떤 컴퓨터나 운영체제, 웹 브라우저를 사용하든지, 어떠한 환경에 처해 있든지 구애받지 않고 접근할 수 있는 콘텐츠를 제작할 수 있도록 지침을 제정한다. 웹 저작도구는 개발자가 접근성 있는 콘텐츠를 생성해 내는 저작도구를 설계하도록 돕고, 개발을 지원한다. 사용자 에이전트 지침에서는 사용자 에이전트를 제작하기 위해 개발 시에 준수해야하는 요구사항을 기술하고 있으며, 범용 운영체제가 탑재된 데스크톱 컴퓨터나 노트북 컴퓨터에서 범용 콘텐츠를 처리하도록 설계된 사용자 에이전트를 말한다. 표준화 작업은 장애인, 노인 등 모두가 함께하는 인터넷 환경을 구축하기 위해 필요한 작업이다.

2.1.4 국내 장차법의 단계적 범위

인터넷은 전자정부, 인터넷뱅킹, 전자상거래, 교육, 뉴스 등 일상생활의 필수도구로써 장애우를 포함한 모든 사람들에게 가장 중요한 생활 수단이 되었다. 현재 장애우의 인터넷 이용률은 52% 정도로, 전체 국민 77%와는 많은 격차가 발생하고 있고, 이러한 차이를 감소시키고자 공공기관에서는 매년 개선 사항을 보이고 있다. 2008년 기준으로 중앙행정기관과 광역자치체, 입법사법기관은 각각 90.6%, 91.6%, 89.5%로 우수하나 기초지방자치단체와 전자정부의 경우에는 각각 83.3%, 80.3%로 다소 미흡한 실정이며, 일반 민간 사이트의 경우는 표준화가 전혀 진행되지 않아 웹 접근성이 매우 낮은 실정이다[12]. 2009년부터 장차법이 의무적으로 시행됨에 따라 정부기관과 특수학교, 종합병원의 표준화는 단계적으로 이루어져야 한다.

장차법의 단계적 시행 범위[20]를 살펴보면 <표 1>과 같다.

<표 1> 국내 장차법의 단계적 범위

행위 자 기간	공공 기관	교육기관 (책임자)	의료기관	복지시설	문화예술 체육	민간사업 장,노동 조합
1년 이내 (‘09)	공공 기관	국공사립 및 특수학교, 특수학급이 설치된 국공립학교, 장애인담당보육시설	종합병원	사회복지시설 (사회복지관 등) 장애복지시설 (요양 및 재활 시설 등)		근로자 300인 이상
2년 이내 (‘10)					국립문화예술 단체, 박물관, 미술관 국립중 앙도서관/미술 관	
3년 이내 (‘11)		국공립유치원 초중고대학교 보육시설 (100인 이상)	일반병원, 치과, 한방병원 (입원 30 인 이상)			근로자 100 ~ 300인
4년 이내 (‘12)					민간종합공연 장	
5년 이내 (‘13)		사립유치원, 평생교육시설, 연수기관, 직업훈련기관 (1000m이상), 보육시설 (100인 이상)	그외 병원 (입원 30 인 이하)		체육관련 행위자	근로자 30 ~ 100인
7년 이내 (‘15’)					민간종합공연 장, 소공연장 (300석 미만) 사립박물관, 미술관	

2.2 전자결재시스템 개요

전자결재시스템은 자동화 이전의 종이 형태로 작성된 문서를 수작업으로 처리하는 방식에서 벗어나 기업 내·외부에서 필요한 문서를 전자 양식을 통해 작성한 뒤 웹상에서 결재하는 시스템으로 시·공간적 제약을 받지 않고 빠른 업무 처리를 가능하게 해준다.

2.2.1 전자결재시스템의 개념

전자결재시스템이란 컴퓨터의 근거리 통신망(LAN)을 통해 문서를 상신하면 결재권자가 컴퓨터상의 전자문서에 결재 처리하는 사무자동화(OA: Office Automation) 시스템의 하나로써 그룹 내에서 신속한 의사결정을 내릴 수 있도록 지원하고, 각 개인 또는 부서 간에 컴퓨터를 이용하여 기안, 보고, 기타 메모 등의 각종 문서를 작성하여 결재를 할 수 있는 기능을 제공한다.

전자결재시스템은 그룹웨어와 인트라넷의 핵심 분야로써 수작업 위주의 사무관리 제도를 개정하여 자동화·전산화 체계로 전환하고 사무 처리의 간소화, 표준화 및 과학화에 필요한 여러 사무관리 요소를 정비한다. 또한 전자문서 등 특수매체 기록물을 공문서의 범위에 포함시키고, 전자서명에 의한 결재문서도 공문서로 간주하며 전산망 유통문서에는 전자 관인을 사용할 수 있도록 한 것으로 문서나 기안을 작성하기 위한 문서작성 기능, 전달하기 위한 전자우편 기능, 전자결재 및 이외 효과적 통제를 위한 응용 기능 등으로 구성되어 있다[21].

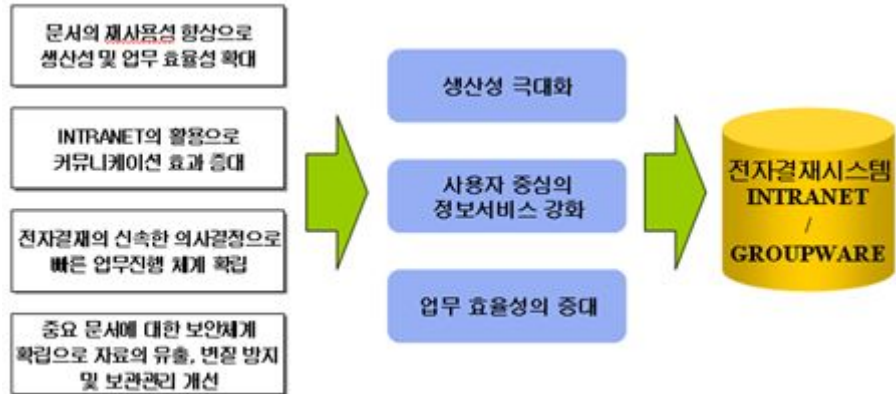
전자결재시스템은 업무의 범위에 따라 다양하게 나눌 수 있는데 그에 따른 정의[22]를 살펴보면 <표 2>과 같다.

<표 2> 전자결재시스템의 정의

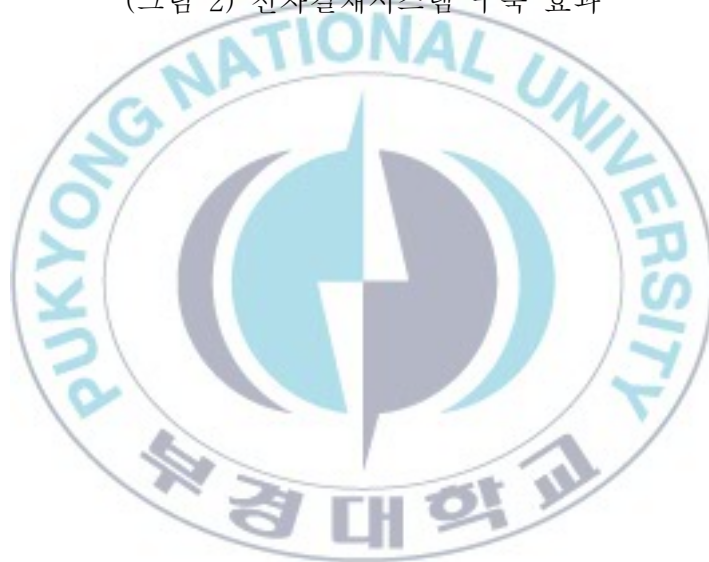
구 분	내 용
협의의 전자결재	결재용 워드문서를 전자우편을 이용하여 송수신 함
일반적인 전자결재	결재전용 전자우편을 구축하여 지정된 수신처로 유통 결재진행관리 및 단순 후속처리를 함
광의의 전자결재	양식작성기에서 결재문서를 이용하여 주요 데이터를 추출 처리하 고, 결재 흐름을 문서에 맞도록 조정, 결재처리 시 유통은 워드 문서 형태로 용도에 따라 변형 처리함

2.2.2. 전자결재시스템의 구축 목적

기존의 문서처리 방식은 서면결재에 따른 서류 증가와 결재를 위한 대기 시간의 지연이 발생하였으며 수작업으로 인한 문서 검토, 수정, 보존을 위한 물리적, 공간적 비용이 증가하였다. 또한 문서 발송, 접수에 따른 시간과 공간의 제약 및 인력 낭비와 서로간의 정보 공유 활용에 많은 어려움이 따랐다[23]. 이런 불편함을 없애고자 도입한 전자결재시스템은 네트워크상에서의 업무 처리로 인해 본사 및 지사, 사내·외간의 신속한 의사 결정 및 처리를 가능하게 해준다. 동일한 인터페이스를 제공하기 때문에 시간과 장소에 구애받지 않고 조직의 핵심 지적 자산인 문서들에 대한 통합 관리로 인해 문서 유출 위험을 최소화 시키며, 관리에 따른 비용을 절감할 수 있다. 문서의 중복을 피할 수 있으며, 빠른 정보검색 및 과거 정보의 재활용으로 업무처리 시간을 단축할 수 있고, 전체 업무 프로세스의 진행 상태에 대해 파악이 가능하며 조정 및 통제도 가능해진다. 기업에서 직원간의 업무효율성 및 생산성을 증대시킬 수 있고, 사용자 중심의 정보서비스를 강화할 수도 있다. 궁극적으로는 그룹웨어 또는 인트라넷 내에서 필수한 구성 요소로 사용되어 편리한 업무 환경을 제공해 준다.



(그림 2) 전자결재시스템 구축 효과



III. 전자결재시스템의 설계

본 장에서는 전자결재시스템 설계 시의 고려 사항 및 업무 처리과정에 대해 살펴보고, 메뉴 및 화면 레이아웃, 데이터베이스의 구조를 직접 설계해 본다.

3.1 전자결재시스템의 설계 방향

전자결재시스템은 종이문서가 가지고 있는 시간적·공간적 제약에 따른 비효율과 불편을 없애고, 네트워크상에서 통용될 수 있도록 문서를 데이터화한 시스템으로 기업의 업무를 통합하는 정보시스템인 인트라넷 혹은 그룹웨어 내에서 문서 결재를 처리하기 위해 사용되는 한 부분이다.

현재 웹 환경에서 구동 중인 사이트는 장차법이 시행됨에 따라 공공기관을 비롯하여 일반 기업에 이르기까지 점차 표준화 작업이 진행되고 있지만 웹 어플리케이션에 대한 부분은 불특정 다수가 아닌 해당 기업에 속해있는 소수만이 사용하는 부분이라는 인식이 강해 표준화 작업이 등한시 되고 있는 실정이다.

비(非) 표준화로 구축되어 있는 웹 어플리케이션의 경우는 브라우저 종류나 버전 차이로 인해 보는 사용자로 하여금 오류가 난 화면을 제공해 줄 수 있고, 마우스를 사용하지 못하는 사용자는 접근이 쉽지 않을 수도 있다. 개발자는 사용자들이 느끼는 불편한 문제점을 해결하기 위해 표준화 작업을 진행할 수 있는데 웹 어플리케이션을 표준화하기 위해서는 프로그램을 변경하기보다는 화면에 보여 지는 디자인 요소를 변경해야 한다.

3.1.1 설계 시 고려사항

웹에서 서비스되는 만큼 모든 콘텐츠는 누구나 쉽게 인식할 수 있도록 설계되어야 하며, 모든 요소들의 기능은 사용자들의 편의성을 고려하여 단순한 형태로 구성되어야 한다. 또한 사용자의 요구사항을 분석하여 현실성에 목적을 두고, 운영환경은 기업 내의 LAN을 활용할 수 있도록 웹을 근간으로 하여 시간적, 공간적 제약을 받지 않도록 응용프로그램으로 구현한다.

세부 콘텐츠는 웹 표준을 적용하여 브라우저의 종류, 버전 등에 관계없이 사용될 수 있어야 하며 키보드만으로도 콘텐츠 접근이 가능해야 한다. 또한 계층적 구조로 사이트를 구성하며, 정보를 지속적으로 등록·수정함으로써 보다 정확하고 풍부한 자료를 관리할 수 있도록 설계되어야 한다. 결재 진행 상태에 대한 추적과 결재 완료된 문서에 대한 검색도 가능해야 한다.

기술적인 측면으로는 W3C의 웹 표준 스펙인 HTML 4.01, XHTML 1.0, CSS, DOM, ECMAScript의 표준안에 따라 제작하며, XHTML 및 CSS를 이용하여 구조와 표현을 분리한다. 웹 브라우저 호환성을 만족시키기 위해 CSS 레이아웃으로 디자인하여 수정과 관리를 용이하게 하고 개발언어로는 ASP, MS-SQL을 이용해서 프로그램화 한다.

3.1.2 업무 처리 과정

전자결재시스템에서의 업무 처리 과정은 (그림 3)에서 보는 바와 같이 관리자는 별도의 관리자페이지에 로그인하여 사원에 대한 기본 정보 및 전자결재에 필요한 기안 양식 정보를 등록한다. 사원은 지정된 아이디로 로

그인한 후 기안할 문서를 선택해서 작성하면 되는데 이때 미흡한 문서는 별도의 임시보관함에 보관하여 추후 재사용할 수 있어야 한다. 작성이 완료된 문서는 상위 결재자에게 상신하고 문서 상신 후에는 결재라인의 흐름에 따라 결재 처리가 이루어지는데 상위 결재자는 권한에 따라 반려, 대결, 전결, 결재 처리를 할 수 있게 된다. 문서하자 등의 이유로 반려된 문서의 경우에는 반려 사유를 반드시 적어 최초 기안자에게 다시 보내고 수정 후 재 결재 처리를 할 수 있게 한다. 최종 결재가 완료된 문서는 관리함에 별도로 보관되며 각 사원별 문서의 권한에 따라 열람이 가능하게 되고, 최종적으로 실제 업무에 반영된다.



(그림 3) 전자결재 업무 흐름도

3.2 전자결재시스템의 설계 구조

전자결재시스템을 관리자 및 사원에 맞게 구현하기 위해 각 사용자 중심의 메뉴를 구성해 보고 화면 레이아웃 및 데이터베이스의 구조를 설계해 본다.

3.2.1 메뉴 구성

전자결재시스템 설계 시 관리자 및 사용자에게 필요한 페이지를 각각 제공하여 관리자가 로그인한 경우에는 사원에 대한 정보 및 결재 문서 양식을 관리할 수 있는 기능을 주고, 사용자가 로그인 한 경우에는 관리자가 미리 등록해 놓은 회사 사내문서양식을 이용하여 온라인상에서 문서 결재를 진행하고 관리할 수 있게 메뉴를 구성한다.

<표 3> 전자결재시스템의 주요 메뉴

구분	항목	설명
공통	로그인	관리자 및 사용자로 나누어 각각의 업무에 맞게 기능이 수행되도록 한다.
관리자	부서관리	회사 내에 존재하는 전체 부서에 대한 정보를 관리한다.
	직위관리	회사 내에서 사용 중인 직위 명칭을 관리한다.
	사원관리	전체 사원의 개인 신상정보를 등록한다.
	양식함 관리	각 문서들을 유사한 양식별로 통합하여 관리한다.
	문서양식 관리	회사에서 사용하는 기안서 등의 문서양식들을 분류하여 등록한다.
사용자	문서기안	출장, 휴가 등의 결재를 받기 위해 최초 기안자가 문서를 작성하는 단계이다.
	결재할 문서	기안자가 문서를 상신했을 경우 결재라인에 따른 결재자가 문서를 열람 및 결재/전결/반려 처리를 할 수 있

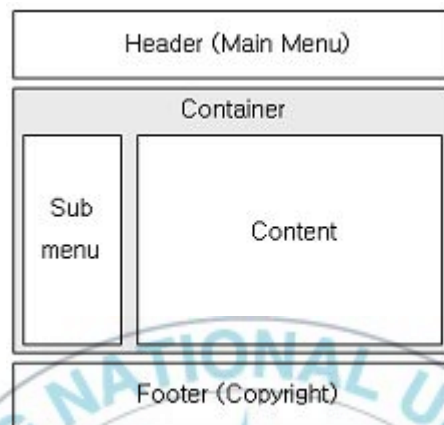
사용자		는 단계로써, 결재자는 결재 문서가 본인의 차례에 왔을 경우에 기안서를 확인할 수 있다.
	결재진행 문서	최초기안자 및 결재라인에 포함된 결재자들이 현재 자신에게 해당하는 문서 상태 및 현황을 열람할 수 있다.
	반려함	결재할 문서가 내용 부족 또는 하자가 있을 경우 현 결재자는 최초기안자에게 반려사유를 적어 문서를 되돌릴 수 있다. 반려된 문서는 최초기안자가 수정을 거쳐 재상신할 수 있다.
	임시보관함	최초기안자가 차후 문서를 상신하기 위해 기안서를 작성하고 저장만 해놓은 상태이다.
	결재완료함	최종 결재자가 결재를 완료했을 경우 해당 문서를 별도로 보관한다.
	수/발신함	최초기안자가 문서를 작성할 때 회사/부서/개인의 형태로 열람권한을 설정하면 결재가 완료된 문서의 경우, 해당 열람자는 문서를 열람할 수 있다.
	환경설정	본인의 개인신상정보를 변경하거나 결재 시 필요한 비밀번호 또는 대리결재자를 직접 설정할 수 있다.

3.2.2 화면 레이아웃 설계

모든 자료들을 통합적으로 관리하도록 각각의 공통요소와 특수요소를 분류하고 특색에 맞게 등록, 검색할 수 있도록 화면을 설계한다. 시스템의 주 기능이 문서처리이므로 텍스트 위주의 간결한 구조로 표시하고, 플래시 등의 동적 환경을 사용하지는 않는다. 또한 콘텐츠를 효과적으로 제공할 수 있는 UI(User Interface)로 구성하고, 순차적 접근 구조를 가진 형태로 설계하며, 이미지 사용을 최소화한다. 탭(Tab)키를 이용한 접근을 가능하게 하여 메뉴와 메뉴 사이를 키보드만으로도 이동이 가능하게 한다.

웹 표준에 맞게 프레임의 사용을 자제하며 반복적으로 사용되는 공통된 콘텐츠의 경우에는 별도의 모듈형태로 만들어 저장한 뒤 각 문서 내의 필

요한 부분에 인크루드 파일(include file) 형식을 이용하여 해당 문서를 불러와서 사용한다.



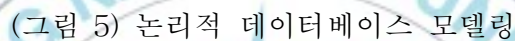
(그림 4) 전자결재시스템의 기본 화면 레이아웃

기본 화면 레이아웃은 크게 Header, Container, Footer의 3단 구조로 나뉘고, Container(sub menu, content)는 가변 폭 2단 컬럼으로 구성하여 주 메뉴의 선택에 따라 자동으로 변경되게 한다.

3.2.3 데이터베이스 설계

데이터베이스는 데이터를 다루는 방식에 따라 관계형, 계층형, 네트워크형, 객체지향형으로 구분할 수 있다. 본 연구에서는 데이터간의 정보를 2차원 테이블 형태로 표현하기 위해 관계형 데이터베이스 이론에 입각해서 모델링하였다.

ER-WIN을 이용하여 설계한 전자결재시스템의 논리적 데이터베이스 모델링은 (그림 5)와 같다.



(그림 6)에서는 전자결재시스템의 핵심이 되는 문서 등록 테이블을 살펴본다.

2: 'ifo_sanction' 테이블 디자인('nts_intranet' 데이터베이스)

	열 이름	데이터 형식	길이	Null 허용
PK	num	int	4	
	id	varchar	20	
	username	varchar	20	
	posino	varchar	10	
	startdate	varchar	20	
	lineid	varchar	255	
	linename	varchar	255	
	viewchk	varchar	10	
	viewid	text	16	✓
	viewname	text	16	✓
	subject	varchar	255	
	content	text	16	
	upfile	varchar	50	✓
	hit	int	4	
	reg_date	datetime	8	
	edit_date	datetime	8	✓
	report	varchar	20	✓
	joblist	varchar	10	✓
	signok	varchar	10	✓
	signall	varchar	10	✓
	signre	varchar	10	✓
	signreid	varchar	20	✓
	signrecnt	text	16	✓
	signsum	varchar	10	✓
	signid	varchar	20	✓
	signidchk	varchar	5	✓

(그림 6) 문서 등록 테이블

문서등록 테이블은 각 결재 문서를 구분하기 위한 결재문서코드, 기안서를 작성하는 주체인 사원에 대한 정보(사원ID, 사원명, 부서코드) 및 결재 관련 정보(결재라인코드, 결재라인명, 결재진행상태, 결재완료 수, 전결유무, 반려유무, 반려자ID, 반려사유, 결재자 수 결재자 ID, 결재자체크), 기안 문서 정보(문서일련번호, 업무보고일, 제목, 내용, 첨부파일, 조회 수, 등록일, 수정일), 수신처 정보(수신처 구분, 수신처 ID, 수신처명)들로 구성되며 각 컬럼들은 데이터에 맞는 형식과 길이를 가진다.

IV. 전자결재시스템의 구현

본 장에서는 전자결재시스템의 디자인 및 프로그램을 웹 표준 형식에 맞게 구축해 보고 기존 시스템과의 차이점에 대해 살펴본 뒤 그 결과를 확인해 본다.

4.1 개발 환경

하드웨어의 환경은 TCP/IP 상에 연결된 서버와 클라이언트 형태로 구성되며, 클라이언트가 서버 측에 자료를 요청하면 서버는 데이터를 가공 처리하여 그 결과를 브라우저를 통해 클라이언트에게 전달한다. DB 서버는 데이터 서버와 별도로 두어 트래픽 및 보안성을 향상시키고, 서버 간의 연동은 OLEDB를 이용한다. 개발자는 Web GUI 개발 툴 및 ASP 모듈을 Windows Server에 주기적으로 업로드하여 사용자의 편의성을 높인다.



(그림 7) 전자결재시스템 하드웨어 구성도

본 연구에서 사용할 하드웨어 구성 환경은 <표 4>와 같다.

<표 4> 하드웨어 구성 환경

구분	사양
중앙처리장치(CPU)	IBM XSERIES_3500 Intel(R) Xeon(R) CPU E5335 @ 2.00GHz
주기억장치(RAM)	1024MB
보조기억장치(HDD)	160G

본 연구에서 사용할 소프트웨어 구성 환경은 <표 5>과 같다.

<표 5> 소프트웨어 구성 환경

구분	사양
운영체제	Microsoft Windows XP Professional SP2
웹 서버	Windows Server 2003
DBMS	MS-SQL 2000
저작 언어	(X)HTML, JavaScript, ASP, CSS
웹 브라우저	Internet Explorer 6.0/7.0, Mozilla FireFox, Opera, Google Chrome, Apple Safari, Netscape Navigator
웹 에디터	editplus v3.01, Dreamweaver CS
통신 프로토콜	HTTP, TCP/IP

4.2 디자인 구현

웹 어플리케이션의 접근성을 높이려면 사이트는 표준 기술로 개발되어야 한다. 현재의 웹 콘텐츠들은 기존의 전통 컴퓨터 형식에서 벗어나 이기종의 브라우저 및 PDA 등 여러 가지 장치들에 의해서 제공되어 지는데 이런 환경들로 인해 HTML은 화면상에 부정확한 정보를 표시하게 되었다. 접근성을 높인 표준화 단계에서는 이런 HTML의 문제점을 보완하기 위해 확장성, 이식성이 좋은 XHTML을 이용하는데 XHTML은 HTML과 코드가 거의 비슷하지만 XML의 형식을 따르고 있기 때문에 문서의 최상단에는 ‘<!DOCTYPE’으로 시작해야 하는 과정이 있다. 모든 XHTML 페이지에서는 문서의 유형이 무엇인지, 어느 표준을 따르는지에 대해 정의가 필요하며 doctype의 결여나 부정확한 사용은 웹 브라우저가 CSS 기반의 웹 페이지를 잘못 표현하게 만드는 원인이 된다.

웹 접근성에 의한 사이트를 구축하는 데는 크게 문서 표준 정의, 영역 범위 및 이동 경로 지정, 웹 표준 적용의 3단계로 나눌 수 있다.

4.2.1 문서의 표준 정의

사이트 제목 등 생성하고자 하는 문서의 기본 정보를 정하고, 문서형 선언 및 meta정보를 입력하여 사이트의 경로 및 출처를 밝힌다. XHTML 최상단에는 문서의 형식을 정의하고, <meta> 태그에 문서의 정보를 기록한다. charset에는 문서의 문자 인코딩 정보를 지정하고, keyword에는 검색로봇이 검색하기 위해 필요한 키워드를 지정한다. keyword는 검색사이트에서 상위 순위로 표시되는 것에 영향을 주는 중요한 부분이다.

<title> 태그는 제목을 입력하는 부분으로써 해당 문서를 나타낼 수 있는 핵심 단어를 입력한다. utf-8 인코딩 문서에서 head 태그 맨 상단에 한글이 포함된 title이 오면 IE 일부에서는 문서가 하얗게 보여 지는 현상이 일어나기 때문에 head 태그 맨 상단에 title을 선언하기 보다는 meta 태그 인코딩 후에 선언한다.

표준화 문서의 기본 정보를 살펴보면 (그림 8)과 같다.

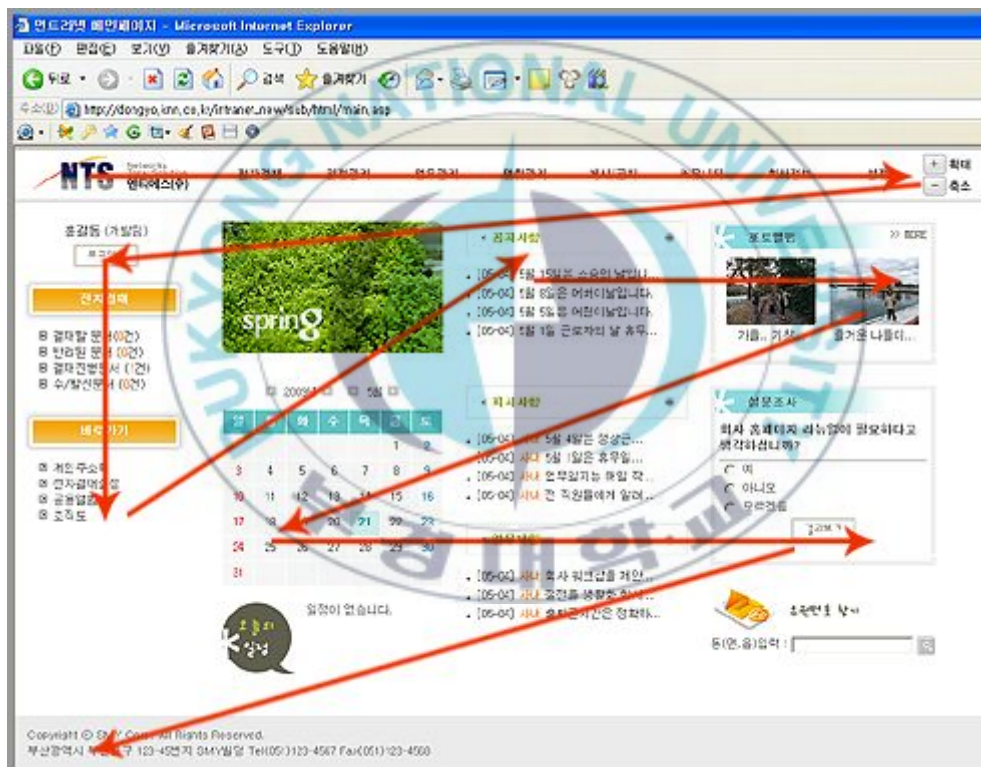
```
//문서 표준 정의
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    //현재문서를 한글로 작성
    <meta http-equiv="Content-Type" content="text/html; charset=euc-kr" />
    //검색로봇이 검색하기 위한 키워드 입력
    <meta name="keyword" contents="전자결재시스템" />
    //해당 페이지의 내용을 요약
    <meta name="description" contents="전자결재시스템 내용입니다" />
    <meta name="author" contents="서미영" /> //제작자
    <meta name="copyright" contents="(c)intranet.com" /> //저작권
    <meta name="replay-to" contents="abcde@gmail.com" /> //메일주소
    <meta name="data" contents="2009-01-01" /> //제작일
    <title>경비지출 내역서입니다</title> //문서의 제목
  </head>
  <body>

</body>
</html>
```

(그림 8) 표준화 문서의 기본 정의

4.2.2 영역 범위 및 이동 경로

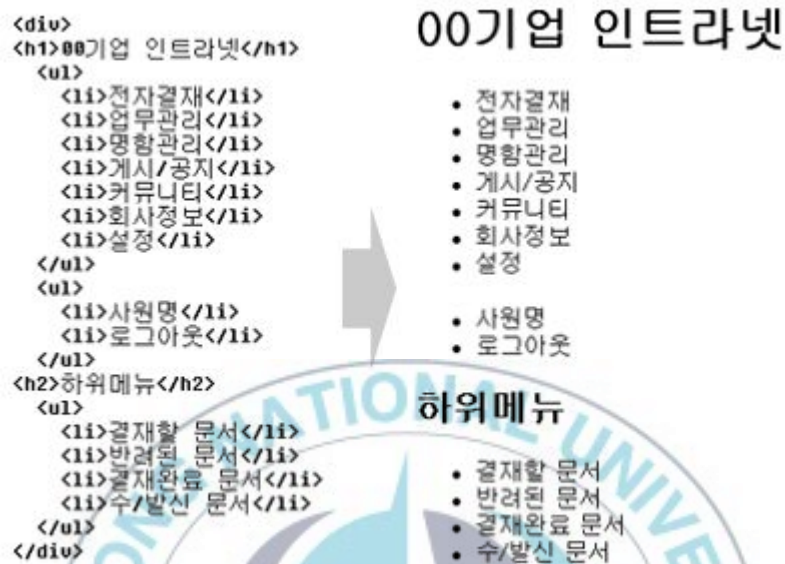
일반적으로 사용자들이 사이트를 볼 때 시각적으로 움직이는 동선 경로는 좌에서 우, 위에서 아래로 흐름을 알 수 있다. 이는 이전의 수많은 사이트가 3단 구조의 형식을 띄고 있었고, 사용자들이 이미 유사한 형식의 사이트 구조에 많이 노출되었기 때문에 익숙해진 까닭도 있다.



(그림 9) 콘텐츠의 시각적 이동 경로

전체적인 레이아웃의 기본 영역 범위는 사용자의 시각적 경로에 따라 주 메뉴 - 하위메뉴 - 콘텐츠 - 카피라이트 순으로 나열한다. 기본 구조가 완

성되면 프로그램을 코딩하고, 스타일시트를 이용하여 디자인을 입힌다.



(그림 10) 계층형 문서 구조

웹 표준으로 코딩된 문서는 계층형 구조로 표현되며 실행된 결과를 살펴 보면 순차적으로 콘텐츠가 나열된 것을 확인할 수 있다. HTML에서는 구조화된 마크업만을 사용하고 모양이나 디자인에 관한 것은 CSS로 완전히 분리하여 사용한다.

4.2.3 웹 표준화 적용

기본 레이아웃에 웹 표준을 적용하여 최종적으로 화면을 구성하는 단계이다. 웹 사이트에 페이지 별로 표현될 콘텐츠들을 나열하고, 구조적 마크업을 사용해서 CSS가 없이도 의미 전달이 가능하게 처리하며, 각 모듈 별로 순차 접근이 가능하게 한다. 텍스트의 경우에는 중요 사항은 앞부분에

서 강조하고, 중요 단어에 링크를 제공하며, 그래픽의 경우에는 파일의 크기를 축소하고, 이미지에 대한 문자 설명을 첨부한다. 클라이언트의 환경을 고려하여 여백의 효과도 조정해야 하고 구조가 동일한 코드는 공통 모듈을 사용하여 소스의 재사용성을 높인다.

(1) 기본 레이아웃 설정

사이트의 레이아웃을 잡을 때에는 기존 <table> 태그를 사용한 형태가 아니라 <div> 태그를 이용하여 구조적, 순차적으로 작업한다. 또한 문서를 전달하기 위한 사이트의 특성상 비주얼적으로 꾸미는 용도가 아니기 때문에 플래시 의존도를 없애고 이미지를 간소화하며 텍스트 위주로 구성한다.

(그림 4)의 기본 레이아웃을 코딩하면 (그림 11)과 같다.

```
<div id="wrapper"> // 전체 Layout을 감싸는 부분
  <div id="topgroup"></div> //Header 부분
  <div id="middlegroup"> // Container 부분
    <div id="leftgroup"></div> // Sub Menu 부분
    <div id="centergroup"></div> // Content 부분
  </div>
  <div id="footergroup"> // Footer부분
</div>
```

(그림 11) 전자결재시스템의 기본 레이아웃

반복되는 링크가 많은 페이지를 효율적으로 이용하기 위해서는 accessibility 기능을 사용한다. accessibility를 <body> 태그 위에 위치시키고 href에 CSS에서 적용한 본문 내용 위치 값을 입력하면 해당 메뉴로 바로 갈 수 있다. Tab 키를 눌렀을 때 포커스 이동순서를 정할 수 있게 키보드 컨트롤도 설정한다.

```

<div id="accessibility">
  <ul>
    <li><a href="#leftgroup" title="서브메뉴 바로가기">서브메뉴 바로가기</a></li>
    <li><a href="#centergroup" title="콘텐츠 바로가기">콘텐츠 바로가기</a></li>
  </ul>
</div>

```

(그림 12) 반복 메뉴 건너뛰기

(2) 텍스트 표현

문서의 제목과 계층을 나타내기 위해서는 h1 ~ h6를 사용하는데 <h> 태그는 단순 제목만 의미하는 것 외에 검색엔진에서 우선적으로 검색되는 기능을 가지고 있다.

글자의 굵기를 나타낼 때 보다는 을, 기울임을 나타낼 때 <I>보다는 을 사용한다. 이는 문서 모양을 나타내는 것보다 명령어를 통해 문서의 의미와 구조를 전달하기 위함이다. 은 강조(emphasis)를 나타내고 은 강한 강조(stronger emphasis)를 나타내는데 시각적 사용자 에이전트는 일반적으로 텍스트를 이탤릭체로 표시하고 텍스트를 굵은체로 표시하는 반면 음성합성기 사용자 에이전트는 볼륨, 높이, 속도 등의 음성합성 파라미터를 태그별 다르게 적용한다. 시각적·음성적으로 CSS를 사용해 텍스트를 굵은체나 이탤릭체로 지정할 수 있으므로 문서구조와 모양을 분리하는 것이라면 과 을 사용한다.

텍스트를 문단으로 정의할 때는
을 통한 줄 바꿈을 지양하고 의미에 맞게 <p>태그를 사용한다. 텍스트를 강조하고자 할 경우에는 색채를 사용할 수 있는데 본문과 핫 워드의 색상이 확연히 구별되어야 하고, 전체적인 글 읽기를 방해하지 않아야 하며 배경과 문자가 구별되어야 한다.

(3) 이미지 표현

텍스트가 아닌 콘텐츠에는 모두 대체 텍스트(alt="이미지설명")가 제공되어야 하며 정보가 있는 콘텐츠는 배경이미지로 사용하지 않아야 한다.

이미지에 연결 링크나 마우스 오버/아웃의 효과를 주기 위해서는 앵커(a) 태그를 사용한다. 앵커 태그에 CSS 효과를 줌으로 마우스의 상태에 따라 다양한 이미지를 보여줄 수 있다. 유사 클래스가 올바르게 동작하기 위해서는 link → visited → hover → active 순서로 정의한다. interlaced GIF 이미지 및 preload 이미지를 사용하여 페이지 로딩 속도를 빠르게 한다.

(4) 콘텐츠 구성

div 등 HTML의 각 태그들은 id나 class 값을 사용하여 각각에 대한 의미를 부여하고, 모듈 별로 관리하며 CSS를 통해 크기, 색상, 폰트 등 제어가 가능하게 처리한다. 순서가 있는 목록은 , 순서가 없는 목록은 , 정의형 목록은 <dl></dl>을 사용한다. 또한 페이지 내에서 과도하게 iframe을 사용하지 않고 프레임 별로 적절한 타이틀을 제공한다. 웹 표준에서는 시작 태그가 있으면 반드시 종료 태그가 있어야 하며, 단독으로 사용되는 태그일 경우에는 해당 태그 뒤에 /(슬래시)를 넣어 준다.

데이터 테이블에서는 테이블에 대한 summary(요약)와 caption(제목)을 제공하여 의미적으로 이해할 수 있도록 적절히 코드를 사용하고, 테이블 행은 테이블 헤더(THEAD), 테이블 푸터(TFOOT), 테이블 본문(TBODY)으로 구성하며, 표 제목 등 중요한 칸에 굵은 글꼴, 가운데 정렬로 표시할 경우에는 <th> 태그를 사용한다.

(5) 온라인 서식

온라인 서식을 작성할 경우 <label> 태그를 사용한다. alt는 대체 텍스트

트, title은 부가 텍스트로 사용하는데 title은 alt나 text를 설명하는데 부족함이 있을 때 사용한다. 필수 입력되는 부분을 *로 넣었을 경우 스크린리더기에서는 별표로 인식하기 때문에 title을 사용하여 “필수항목”으로 넣는다. label로 지정된 입력 폼은 해당 텍스트를 클릭하면 그에 연관된 입력 폼에 커서가 이동함으로써 선택영역범위가 넓어지는 효과까지 확인할 수 있다. 이는 라디오 버튼이나 체크박스의 경우 <label> 내의 텍스트를 사용자가 클릭할 때 컨트롤 선택을 토글해 줌으로써 클릭할 수 있는 영역이 넓어져서 사용자가 좀 더 쉽게 폼의 값을 선택할 수 있다. tabindex를 지정하면 탭 키를 누를 경우 지정한 순서대로 포커스가 이동하여 모든 동작을 키보드로만 하는 지체 장애우들이 좀 더 쉽게 사이트에 접근할 수 있다.

전자결재시스템의 문서 기안 온라인 양식을 살펴보면 (그림 13)과 같다.

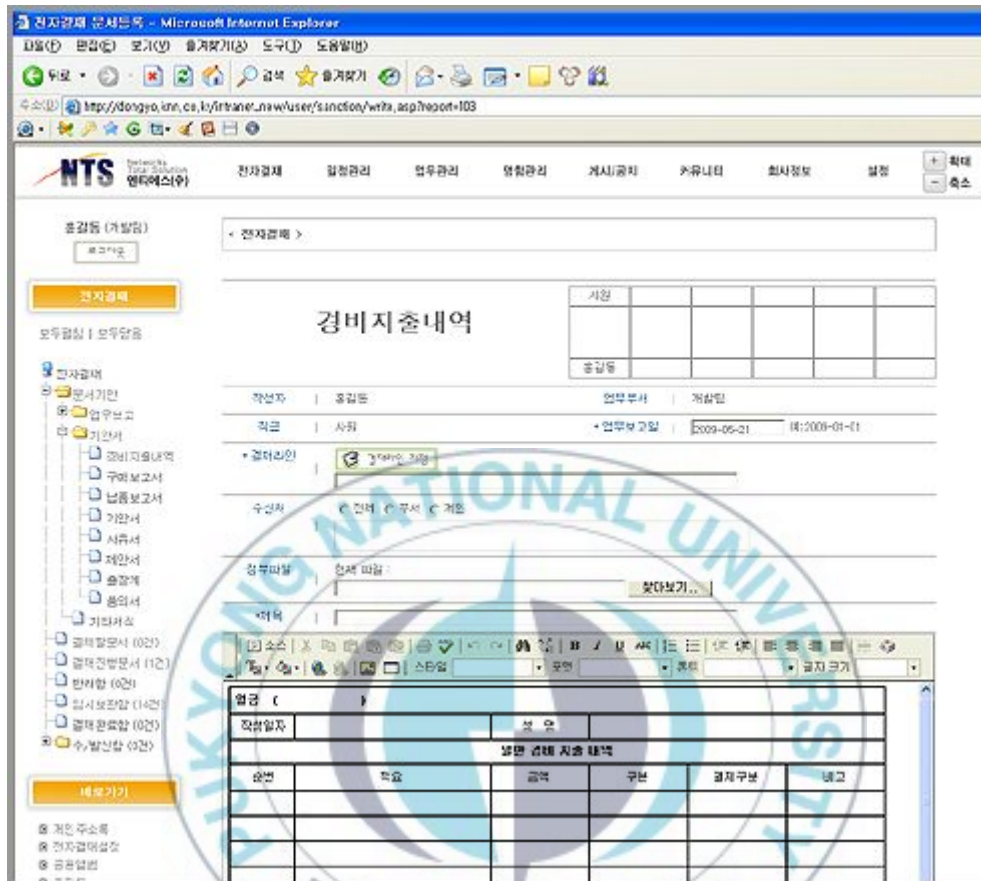
```
<form name="uForm" method="post" action="write_ok.asp">
<div class="contentline">
  <ul>
    <li id="title">작성자</li>
    <li id="content2"><%=Session("username")%></li>
    <li id="title">업무부서</li>
    <li id="content2"><%=PosiChk("posiname")%></li>
    <li id="title">직급</li>
    <li id="content2"><%=pordname1("pordname")%></li>
    <li id="title">
      <label for="startdate"><span title="필수항목">*</span> 업무보고일</label></li>
    <li id="content2"><input name="startdate" type="text" id="startdate" size="15"
maxlength="10" title="업무보고일을 입력"/> 예:2009-01-01</li>
    <li id="title"><label for="linename"><span title="필수항목">*</span>
결재라인</label></li>
    <li id="content1"><a href="javascript:linechk();"></a><br /><input name="linename" type="text" id="linename"
```

```

size="70" maxlength="100" readonly/><input name="lineid" type="hidden"
id="lineid"/></li>
<li id="title">수신처</li>
<li id="content1"><input type="radio" name="viewchk" value="all" id="all"
title="전체인 경우 선택"/><label for="all">전체</label><input type="radio"
name="viewchk" value="posi" onClick="javascript:viewchk1();" id="posi" title="부서인
경우 선택"/><label for="posi">부서</label><input type="radio" name="viewchk"
value="user" onClick="javascript:viewchk2();" id="user" title="개인인 경우 선택"/><label
for="user">개인</label><br />
<textarea name="viewname" cols="70" rows="2" readonly
wrap="PHYSICAL"/></textarea><textarea name="viewwid" cols="70" rows="3" readonly
wrap="PHYSICAL" style="display:none;/></textarea></li>
<li id="title"><label for="subject"><span title="필수항목">*</span>제목</label>
</li>
<li id="content1"><input name="subject" type="text" id="subject" size="70"
maxlength="70" title="제목을 입력"/></li>
</ul>
</div>
<div class="contentwrite">
<% Dim sBasePath, oFCKEditor
sBasePath = "../fckeditor/"
Set oFCKEditor = New FCKEditor
oFCKEditor.BasePath = sBasePath
oFCKEditor.Value = content
oFCKEditor.Create "content" %>
</div>
<div class="userbutton">
<a href="javascript:Check('N');"></a>
<a href="javascript:Check('Y');"></a>
<a href="javascript:cancel()"></a>
</div></form>

```

(그림 13) 전자결재시스템의 문서 기안 소스



(그림 14) 전자결재시스템의 문서 기안 페이지

문서 양식을 보여주기 위한 HTML 에디터는 FCKeditor를 사용하였고, HTML과 스크립트는 웹 표준 형식을 따르고 있다. 이는 어느 브라우저에 서나 동일한 화면과 기능을 제공해 준다.

(6) 스크립트의 표준화

아이디를 사용하여 HTML의 element를 구해오는 방식으로 많이 사용되 는 방법은 document.all 메소드를 사용하는 방법과 document.getElement-

ById를 사용하는 두 가지 방법이 있다. 프로그램 내에서 스크립트를 사용할 경우에는 인터넷 익스플로러 환경뿐만 아니라 이기종의 브라우저에서도 구현될 수 있도록 표준화 작업이 필요하다. 기존의 document.all의 경우에는 익스플로러에서만 동작하는 특수한 메소드이기 때문에 넷스케이프 계열에서는 사용할 수 없다. 대부분의 브라우저를 고려한다면 표준 스크립트인 getElementById를 사용하여야 한다.

기존 인터넷 익스플로러에서만 실행되는 스크립트와 표준 스크립트를 비교해 보면 (그림 15)와 같다.

[인터넷 익스플로러 위주의 스크립트]
<pre>function reChecked(chk){ if(chk=="ok"){ document.all.SelectCrit.style.style.display='block'; }else{ document.allSelectCrit.style.style.display='none'; } }</pre>
[표준 스크립트]
<pre>function reChecked(chk){ if(chk=="ok"){ document.getElementById('contentresign').style.display='block'; }else{ document.getElementById('contentresign').style.display='none'; } }</pre>

(그림 15) 스크립트 비교

전자결재시스템 메인 메뉴를 일반적인 HTML 형태의 코딩한 방식과 웹 표준을 적용한 방식으로 구현한 결과는 (그림 16), (그림 17)과 같다.

[HTML 형식]

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td valign="top" background="images/top_bg.gif">
<table border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="right" valign="top"><a href="center.asp" target="main"
onFocus="this.blur()"></a></td>
<td valign="top">
<table width="756" height="48" border="0" cellpadding="0" cellspacing="0">
<tr>
<td><a href="sanction_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg01','','images/over_m00.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
<td><a href="schedule_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg02','','images/over_m01.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
<td><a href="work_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg03','','images/over_m02.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
<td><a href="address_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg04','','images/over_m03.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
<td><a href="board_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg05','','images/over_m04.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
<td><a href="community_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg06','','images/over_m05.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
        <td><a href="company_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg07','','images/over_m06.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
        <td><a href="setup_idx.asp" target="main" onfocus="this.blur()"
onMouseOver="MM_swapImage('topimg08','','images/over_m07.gif',1)"
onMouseOut="MM_swapImgRestore()"></a></td>
    </tr>
</table>
</td>
</tr>
<tr bgcolor="#e5e5e5"><td height="1" colspan="2"></td></tr>
</table>
</td>
</tr>
</table>

```

(그림 16) 일반적인 HTML 형태의 코딩 방식

[CSS 형식]

```

#topgroup{ width: 100%; height: 60px; background-image: url(images/top_bg.gif); }
#zoom { float: left; margin-top: 8px; }
#logo { float: left; margin: 0; }
#mainmenu ul li { float: left; }

```

[XHTML 형식]

```

<link href="../sub/css/default.css" rel=stylesheet type="text/css">
<div id="topgroup">
<h1 id="logo"><a href="/main.asp"></a>
</h1>
<div id="mainmenu">
    <ul>

```

```

<li><a href=/sanction/list.asp?joblist=sign"></a></li>
<li><a href=/schedule/list.asp?viewchk=total"></a></li>
<li><a href="/project/list.asp?viewchk=all"></a></li>
<li><a href="/address/list.asp?viewchk=all"></a></li>
<li><a href="/notice/list.asp?viewchk=all"></a></li>
<li><a href="/infodoc/list.asp?boardname=N"></a></li>
<li><a href="/cominfo/list.asp?boardname=N"></a></li>
<li><a href="/member/list.asp?boardname=N"></a></li>
<li><div id="zoom">
<a href="javascript:styleZoom(1);"></a><br /><a href="javascript:styleZoom();"></a></div></li>
</ul>
</div>
</div>

```

(그림 17) 웹 표준을 적용한 프로그램 코딩 방식

웹 표준을 적용한 코딩 방식은 비(非)표준화 코딩 방식에 비해 소스가 간결해지기 때문에 상대적으로 파일 크기도 작아진다. 위의 메인 메뉴에 대한 소스 코드의 파일 크기만 보아도 4.26KB에서 2.71KB로 줄어든 것을 확인할 수 있었다. 또한 각각의 항목들은 XHTML과 CSS의 모듈 별로 구분되어 있어 언제든지 재사용 할 수 있다.

4.3 프로그램 구현

웹 표준은 프로그램의 기능적 요소보다는 사용자에게 보여 지는 디자인적 요소에 중점을 두고 구조적으로 설계, 개발되어진다. 단순히 콘텐츠를 보여주는 목적만 있다면 XHTML과 CSS를 이용하여 표준화에 맞게 구현할 수 있지만 문서를 처리하고 열람하는 등 사이트를 동적으로 표현하기 위해서는 별도의 웹 프로그램 언어(ASP, PHP, JSP 등)를 사용해야 한다.

전자결재시스템이 구동되기 위해 필요한 프로그램 중 결재관련 프로세스의 주요 부분을 살펴보면 (그림 18)과 같다.

```
sql="SELECT * FROM ifo_member WHERE id='" & Session("id") & "'"
Set SignPdCk=dbCon.Execute(sql)
if SignPdCk("signchk")="N" then //문서상신일 경우
    if signchk="1" then //기안문서 상신
        sql = "UPDATE ifo_sanction SET joblist='sign', signok='1', signre='N' WHERE num='" & num
        dbCon.Execute(sql)
        Response.Redirect "list.asp?joblist=ing&reing=ok"
    elseif signchk="2" then //결재처리
        sql = "UPDATE ifo_sanction SET signok=signok+1 WHERE num='" & num
        dbCon.Execute(sql)
        Response.Redirect "list.asp?joblist=ing&reing=ok"
    elseif signchk="3" then //전결처리
        sql = "UPDATE ifo_sanction SET signok=signok+1,signall='Y' WHERE num='" & num
        dbCon.Execute(sql)
        Response.Redirect "list.asp?joblist=ing&reing=ok"
    else //반려처리
        sql = "UPDATE ifo_sanction SET joblist='write', signok='', signall='N', signre='Y', signreid='" & signreid & "',signreent='" & signreent & "'" WHERE num="
```

```

& num
    dbCon.Execute(sql)
    Response.Redirect "list.asp?joblist=reply&reing=ok"
end if
else //문서 재상신일 경우
    if signchk="1" then
        sql = "UPDATE ifo_sanction SET joblist='sign', signok='1' ,signre='N' WHERE
num=" & num
        dbCon.Execute(sql)
        Response.Redirect "list.asp?joblist=ing&reing=ok"
    else
        Response.Redirect "signpwd.asp?signchk=" & signchk & "&joblist=" & joblist &
"&num=" & num & "&signreid=" & signreid & "&signrecl=" & signrecl
    end if
end if

```

(그림 18) 전자결재시스템의 결재 프로세스

결재처리 프로세스에서는 최초 기안자가 문서를 상신 시 현재 문서 진행 상태를 결재(sign)로 변경하고, 결재완료 수는 1로 지정한다. 결재라인에 따라 상위 결재자의 결재 처리가 완료될 때마다 결재완료 수는 계속 증가한다. 결재완료 수는 결재라인에 등록된 결재자 수와 비례하여 동일해지면 비로소 결재가 완료된 것이다. 상위 결재자의 부재로 전결처리를 해야 할 경우에는 전결유무를 체크하고, 반려 처리에서는 결재 진행 상태를 문서 작성(write) 중인 형태로 변경한 뒤 결재완료 수를 공백으로 처리하여 수정 후 재결재가 가능하게 한다.

4.4 기존 시스템과의 차이점

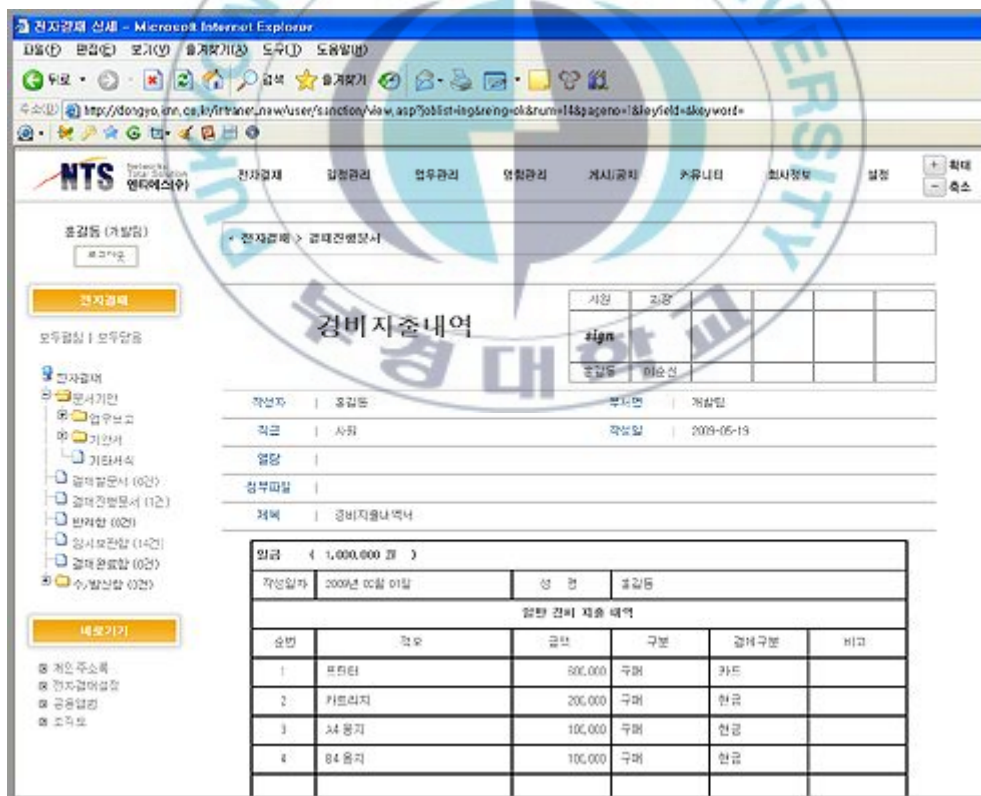
웹 표준을 적용한 코딩 방식은 일반적인 코딩방식에 비해 소스 코드량이 적어지므로 서버에서의 다운로드 속도가 그만큼 빨라지며, 저장 공간도 작게 차지되기 때문에 회선 트래픽도 절약할 수 있다. 그 밖의 기존 코딩 방식과의 차이점을 살펴보면 <표 6>과 같다.

<표 6> 기존 전자결재시스템과의 차이점

기존 웹 환경의 전자결재시스템	웹 표준을 적용한 전자결재시스템
장애우를 배제한 시스템으로 설계되어 마우스/키보드 조작을 통해서만 사이트 접근이 가능	모든 사용자를 고려한 시스템으로 설계(전맹 제외)되어 키보드만으로도 사이트 접근이 가능
Internet Explorer 6.0 웹브라우저 환경에 따른 프로그램 코딩 (브라우저 지원 환경에 따라 이용이 불가능할 수도 있음)	Internet Explorer 6.0/7.0, Firefox, Netscape 등 이기종의 웹브라우저에서도 접근이 가능한 표준 코딩
웹문서는 Html 코드를 이용해서 작성하기 때문에 문서가 생성될 때마다 매번 동일한 소스를 재 코딩해야 하는 불편함이 따름	CSS코드를 사용하여 모듈별로 작성하기 때문에 동일한 모듈은 언제든지 재사용이 가능하고 한 번의 수정처리로 공통 소스를 변경할 수 있으며, 확장/이동이 용이
웹 저장 및 문서 편집 프로그램의 복사, 붙이기 기능 이용 시 html의 모든 소스를 가져올 수 있음	웹 저장 및 문서 편집 프로그램의 복사, 붙이기 기능 이용 시 CSS의 구조적인 정보를 뺀 나머지 텍스트, 이미지 정보만을 가져오기 때문에 콘텐츠 전체 복제 예방이 가능
각 페이지별 소스 코드가 길어짐	페이지별 소스 코드가 간결해짐
사이트 내 이미지 오류 발생 시 텍스트 외 내용확인 불가	이미지에 의존하지 않기 때문에 텍스트(또는 대체 텍스트)만으로도 전체 내용 확인이 가능

4.5 구현 결과화면

기존의 비(非)표준으로 구현된 사이트들은 브라우저의 종류 및 버전에 따라 서로 다른 화면을 보여주었다. 반면 사용자 인터페이스를 강화하여 웹 접근성을 높인 사이트의 경우에는 사용자간의 다른 시스템 환경에서도 동일한 결과를 보여준다. 전자결재시스템에서 작성된 문서를 살펴보면 사이트가 표준으로 구축되었을 경우에는 인터넷 익스플로러 v6.0 이외에도 v7.0, 모질라 파이어폭스, 오페라, 구글 크롬, 애플 사파리, 넷스케이프 내비게이터 브라우저에서 동일한 결과로 보여 진다는 것을 알 수 있다.



사원	과장	승인	비고
김비자	김비자	승인	
김비자	김비자	승인	

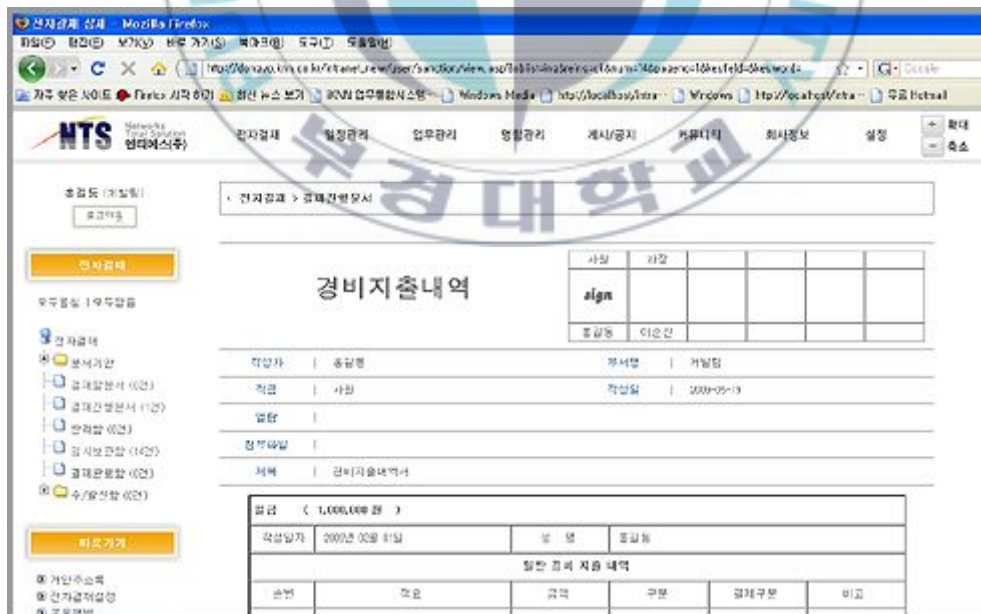
작성일자	작성인	작성일자	작성인
2009-05-19	김비자	2009-05-19	김비자

승인	결재	구분	결재구분	비고
1	프린터	500,000	구매	카드
2	카트리지	200,000	구매	현금
3	A4 용지	100,000	구매	현금
4	B4 용지	100,000	구매	현금

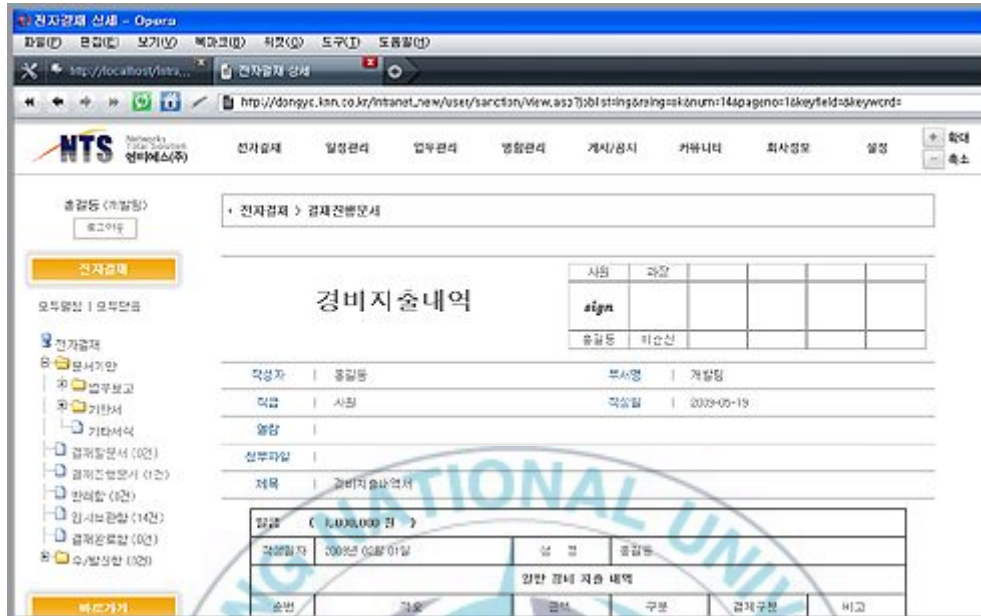
(그림 19) 인터넷 익스플로러(Internet Explorer v6.0)



(그림 20) 인터넷 익스플로러(Internet Explorer v7.0)



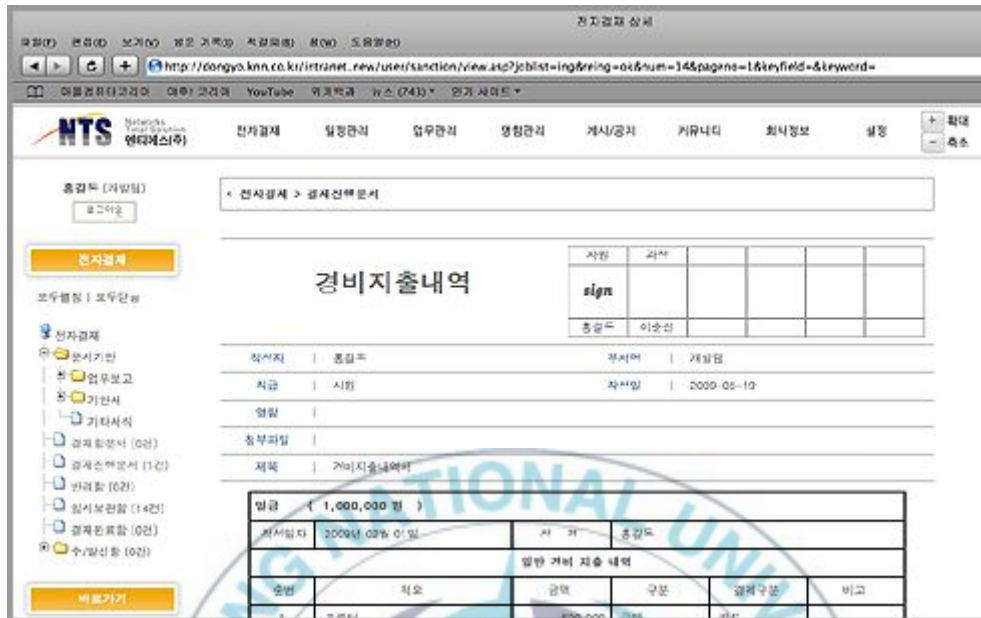
(그림 21) 모질라 파이어폭스(Mozilla FireFox v3.0.10)



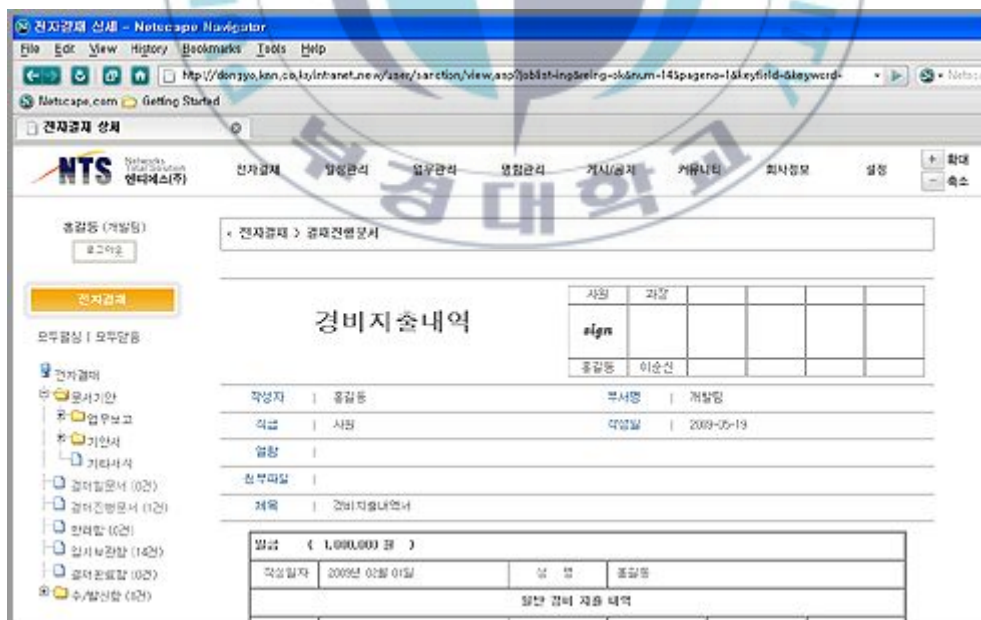
(그림 22) 오페라(Opera v9.64)



(그림 23) 구글 크롬(Google Chrome v0.2.149.30)



(그림 24) 애플 사파리(Apple Safari v3.1.2)



(그림 25) 넷스케이프 내비게이터(Netscape Navigator v9.0.0.6)

V. 결론 및 향후연구

기존의 인터넷 익스플로러 6.0 환경에서 HTML로 구현된 전자결재시스템은 웹 브라우저의 특성에 따라 접근이 불가능하거나 페이지 오류를 유발하였으나 웹 표준을 적용한 전자결재시스템은 인터넷 익스플로러의 최신버전(v7.0) 뿐만 아니라 모질라 파이어폭스, 오페라 등의 이기종 웹 브라우저에서도 동일한 결과로 출력되어 크로스 브라우징이 가능한 것을 확인할 수 있었다.

모듈별 순차적인 형태로 코드를 구성하였기 때문에 마우스를 사용하지 못하는 지체 장애우도 키보드의 tab 키를 사용하여 사이트 내 콘텐츠들 간의 이동이 가능하고, 저시력 장애우나 노령자에게는 페이지의 레이아웃 확대 및 축소 기능을 주어 문서를 보다 쉽게 읽을 수 있는 환경을 제공하였다. 또한 대체 텍스트의 제공으로 이미지에 오류가 발생하였을 경우에도 데이터 확인이 가능할 수 있도록 구현함으로써 사용자에게 높은 접근성과 편의성을 주었다.

본 전자결재시스템은 표준화를 통한 CSS와 HTML의 분리로 인해 사이트의 유지보수 및 관리가 용이하고, PDA, 프린트, 브라우저 등 사용자 각각의 작업 환경에 맞게 구현도 가능하다. 기업에서의 업무 처리 시 장애인의 경우 텍스트 위주로 열람만 가능한 것이 아니라 직접 문서를 손쉽게 작성할 수도 있고 일반 사용자의 입장에서 시스템 환경에 제약을 받지 않기 때문에 업무 처리에 따른 생산성 및 효율성도 기대된다.

향후에는 전자결재시스템을 포함하는 전체 인트라넷 시스템에 대한 접근

성을 높일 것이며, 여러 웹 접근성 평가 도구(KADO-WAH 등)를 통한 구현 결과 검증을 통해 사이트를 보다 더 최적화 시킬 수 있는 방안을 연구할 것이다. 또한 전자결재시스템 특성상 문서제공 방법이 HTML 에디터를 사용하고 있기 때문에 시력장애 1급(전맹)인 경우에는 문서 작성에 불편함이 따르고 있으므로 이 부분에 대한 연구도 계속 진행할 것이다.



참 고 문 헌

- [1] World Wide Web Consortium(W3C), <http://www.w3c.org>
- [2] XHTMLTM 1.0 The Extensible HyperText Markup Language (Second Edition), <http://www.w3.org/TR/xhtml1/>
- [3] Cascading Style Sheets, level 1, <http://www.w3.org/TR/REC-CSS1/>
- [4] Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification, <http://www.w3.org/TR/CSS2/>
- [5] Document Object Model (DOM) Level 1,2,3, <http://www.w3.org/DOM/DOMTR/>
- [6] W3C, Web Content Accessibility Guidelines 1.0, <http://www.w3c.org/TR/WCAG10/>
- [7] W3C(2008), Web Content Accessibility Guidelines (WCAG) 2.0, <http://www.w3c.org/TR/WCAG20/>
- [8] 미 재활법 508조, <http://www.section508.gov>
- [9] Market Share, <http://marketshare.hitslink.com>
- [10] 보건복지가족부, <http://www.mw.go.kr>
- [11] 정보통신 접근성 향상 표준화 포럼(IABF), 웹접근성 표준화, <http://www.iabf.or.kr>
- [12] “웹 접근성 현황 및 정책 방향”, 행정안전부 2009. 4.
- [13] 선원진, “공공부문 온라인 교육 사이트의 웹 접근성 준수실태 및 개선 방안”, 숭실대학교 정보과학대학원 석사학위논문, 2007. 8.
- [14] “2008년도 민간개발자 웹 접근성 전문교육”, 행정안전부, 2008.

- [15] 한경희, “시각장애인의 웹 접근성 향상을 위한 웹 디자인에 관한 연구”, 홍익대학교 대학원 석사학위논문, 2007. 2.
- [16] 한대준, “W3C 권고 웹표준에 의거 유효성 검증을 통한 오류의 유형별 분석”, 국민대학교 비즈니스IT전문대학원 석사학위논문, 2008. 2.
- [17] 이종훈, “웹 접근성 지침에 의한 교육용 홈페이지의 평가 방안 및 분석”, 동의대 교육대학원 석사학위논문, 2008. 2.
- [18] 노석준, “시청각장애 학습자를 위한 접근 가능한 웹기반 교수(WBI) 설계 지침”, 한국교육. 제34권 제1호, p.91-117, 한국교육개발원, 2007. 4.
- [19] Dan Cederholm, 박수만 역, “실용예제로 배우는 웹표준”, 에이콘, 2005.
- [20] “웹 접근성 표준 동향”, 한국정보문화진흥원, 2009. 4.
- [21] 이지영, “비밀문서유통을 위한 보안 요구사항 분석”, 대전대 대학원 석사학위논문, 2004. 2.
- [22] 김성철, “ASP.NET 기반 전자결재시스템 설계 및 구현”, 계명대 산업기술대학원 석사학위논문, 2003. 12.
- [23] 정미숙, “전자결재 및 전자문서 유통의 활성화 방안에 관한 연구”, 영남대 산업대학원 석사학위논문, 2002. 8.
- [24] David Sawyer McFarland, 김지원 역, “CSS 비밀 매뉴얼”, 한빛미디어, 2007.
- [25] 행정안전부, <http://www.mopas.go.kr>

감사의 글

기대 반 설렘 반으로 입학한지도 2년이 지나 어느덧 마지막 5학기를 마무리하는 시점에 왔습니다. 직장 생활을 하면서 스스로를 되돌아보니 부족한 점을 많이 느껴 대학원에 입학하게 되었는데 막상 직장 생활과 학교생활을 병행한다는 것은 결코 쉽지 않은 일이었습니다. 이런 저에게 많은 조언과 격려를 해주신 여러 교수님들과 선·후배님들로 인해 어느덧 졸업이라는 영예를 안게 된 것 같아 지면으로나마 감사의 인사를 드리려고 합니다. 논문이 완성되기 까지 많은 생각과 고민들로 밤을 지새우며 하루하루를 보내다 보니 새삼 공부에 대한 재미를 깨닫게 되었고 나태해진 저를 다시 채찍질할 수 있는 계기가 되었습니다.

입학부터 졸업에 이르기까지 많은 관심과 조언을 아끼지 않으시고 성실함이 무엇인지를 몸소 보여주신 여정모 교수님께 깊은 존경과 감사의 마음을 전하고 싶습니다. 또한 바쁘신 와중에도 질 높은 논문이 될 수 있도록 격려와 세심한 지도를 해주신 박홍복 교수님, 윤성대 교수님, 이경현 교수님께도 진심으로 감사드립니다. 그리고 2년 반 동안 저에게 다양한 가르침을 주신 전산정보학과 교수님들께도 이 자리를 빌려 감사의 인사를 드리고 싶습니다. 졸업 후에도 교수님들의 가르침을 본받아 더욱더 발전하고 큰 사람이 되도록 노력하겠습니다.

대학원 생활 동안 부족한 저에게 많은 조언과 도움을 주신 연구실의 정명희 선생님, 박선희 선생님, 신의정 선생님, 윤종욱 선생님, 조혜정 선생

님, 김규태 선생님, 옥영종 선생님, 정보위 선생님, 정충길 선생님, 김은숙
선생님, 학부생들에게 감사의 마음을 전하고 싶습니다. 또한 즐겁게 학교를
다닐 수 있도록 이끌어 주시고 좋은 추억을 간직하게 해주신 전산정보학과
선생님들에게도 고마움을 전합니다. 학교를 다니는데 많은 편의를 봐주시
고 배려해 주신 (주)iknn 동료 여러분들에게도 감사드립니다.

마지막으로 항상 저를 믿어 주시고 뒷바라지 해 주신다고 고생하신 나의
사랑하는 어머니, 누나의 투정을 다 받아주는 착한동생 호진이, 삶에 활력
소가 되어주는 가족과 친구들에게도 감사의 인사를 드립니다. 그 외 일일
이 다 열거하지는 못하지만 언제나 저에게 많은 격려와 성원을 보내주신
모든 분들에게도 다시 한 번 감사의 마음을 전합니다.
사랑합니다. 존경합니다. 그리고 감사합니다.

2009년 6월
서 미 영