

효과적인 웹사이트 구축을 위한 개발방법론에 관한 연구*

A Study on the Web Site Development Methodology for Effective Web Site

오경묵(Kyung-Mook Oh)**

목 차

1 서론	3.4 구현단계
2 웹사이트 개발을 위한 개발방법론	3.5 홍보단계
2.1 최근동향	3.6 혁신단계
2.2 개발방법론	4 웹사이트 공학
3 웹사이트 개발 절차	4.1 폭포수 모델
3.1 기획단계	4.2 프로토타입 모델
3.2 분석단계	4.3 나선형 모델
3.3 설계단계	5 결론 및 제언

초 록

최근의 웹사이트 개발업무는 점점 복잡, 대형화되어 발전됨에 따라, 개발과 관련된 조직전체를 통제하는 시스템을 완성해 갈 수 있는 프로젝트의 관리기법과 이의 바탕이 되는 개발방법론이 필요하게 되었다. 본 연구에서는 조직적이고 체계적인 웹사이트 개발과 운영이라는 개념 자체가 거의 없는 국내 웹사이트 개발 상황에서 국내여건에 맞는 웹사이트 개발 방법론 연구의 필요성과 타당성을 연구, 분석하였다.

ABSTRACT

Developing the web-based information service sites requires their own developing methodology, because the process of the web sites development become more complicate and Internet surroundings are changing rapidly. They should control all components of the system for effective and efficient web services. However there is little clear concept about systematic development and management of web service sites, in the Korean Internet service sectors, especially in information service sectors. The research analyse current situation and necessity of developing methodology, and suggest new approach for developing new methodology of the web business/service sites.

키워드: 웹사이트 개발방법론, 웹사이트 엔지니어링, 인터넷 비즈니스, 웹서비스

* 본 연구는 과학기술부 지원 '99 숙명여자대학교 연구기반 확충사업에 의하여 지원되었음.

** 숙명여자대학교 정보과학부 조교수

■ 논문 접수일 : 2000년 2월 29일

1 서 론

웹사이트를 처음 구축하던 초기 시절에는 한 두 명의 개발자가 필요한 모든 일들을 스스로 배워 나가며 처리해야 했다. 이들은 스스로가 정보 설계 전문가였으며 그래픽 디자이너이자 편집자였고 프로그래머였다. 그러나 웹사이트는 단순히 문서를 인터넷이라는 새로운 매체를 통해 출판하는 수준을 벗어나 그 이상의 역할을 하는 매체로 발전하게 되었고, 웹 개발자들은 더 많은 요구에 부응해야만 했으며, 웹사이트 개발이 될 때까지 점점 더 많은 인원을 필요로 하게 되었다. 이렇게 되자 복잡하게 발전한 웹을 위해 필요한 모든 분야를 한 개인의 능력으로는 감당할 수 없게 되었으며, 각 분야 별로 고유의 전문가를 필요로 하는 단계에 이르렀다. 이제는 실제 개발에 있어 상호 보완적인 여러 분야의 전문기술을 갖춘 개인의 전문가들로 구성된 팀이 함께 공동작업을 진행해야 하는 경우가 대부분이 되고 있다. 따라서 개발에 참여하는 구성원들은 각기 각 분야의 전문가들로서 그 분야에 전문기술이 있어야 하며, 그와 동시에 개발 구성원들간에 긴밀한 정보 교류와 협력이 작업성과에 있어 지대한 영향을 끼치는 요인으로 작용하게 된다.

실제 도서관 정보서비스 혹은 비즈니스를 위한 웹사이트 개발절차에 있어서도 한 개인의 결정보다는 참여자들의 지혜를 모아 최선의 길을 선택하는 것이 바람직하며, 여러 사람들이 의견을 제시할 수 있는 기회가 부여되어야 한다. 이 때 웹 개발에 참여하는 구성원들간의 의견교환은 대화의 형태로 이루어지게 되나, 원활한 업무수행을 위하여 최종적으로 주요항목을 문서화하는 작업이 업무지식에 대한 기록보관과 노하우의 손실을 막기 위해 필요하게 된다. 실제로 IT 관련 컨설팅

업체인 가트너그룹의 연구보고에서는 “핵심 업무와 대인관계 요소를 중시하는 프로젝트 관리기법을 실행하는 기업들은 전체 프로젝트 수의 30%에 달하는 실패를 줄일 수 있다”고 한다(Gartner Group's IT View 1999.9). 그러나 대부분의 국내 웹사이트 개발과정은 변변한 개발관련 산출 문서 하나 없이 개발과 사이트 운영이 이루어지고 있는 실정이다.

대부분의 유용한 사이트의 경우에는 시간이 흘러감에 따라 그 규모나 복잡도 면에 있어 점점 더 심화되어지는 경향을 보이게 된다. 이 경우 사업의 범위가 확장되거나 사업 목적이 크게 바뀌어 시스템과 영업방식에 있어 큰 변화가 필요한 경우에 개발업체는 기존의 업무 파악이 용이치 못하여 서비스를 중단하거나, 시스템을 아예 처음부터 새로 개발해야하는 큰 어려움을 겪을 수 있게 된다. 이러한 큰 장애가 아니더라도 최근 들어 열병처럼 번지고 있는 인터넷 비즈니스가 활성화 되는 시점에서 충분한 기술을 갖춘 전문인력을 확보하기 위해서는 상당한 어려움을 겪게 되며, 높은 이직을 때문에 충분한 경험을 갖추지 못한 신규인력을 고용하여 재교육해야하는 사례가 빈번히 발생하고 있다. 이러한 현 상황에서 웹사이트 개발의 중심 역할을 수행하는 개발책임자들은 기존 업무자에게는 현실태와 문제점 파악을 용이하게 하여 앞으로의 개선방향을 제시하게 되며, 신규 인력에게는 사업전반에 대해 보다 쉽게 이해할 수 있는 기회를 제공함으로써 신속한 업무 적응이 가능토록 해줄 수 있는 새로운 방법을 모색하게 되었다.

외국의 경우, 이미 개발된 웹사이트에 대해서 BPR(Business Process Reengineering) 하듯이 다시 기획부터 시작해서 설계, 개발을 하거나 웹사이트 평가나 클리닉에 대한 요구가 늘어

나고 있는데 (Gartner Group's IT View 1999.6), 이는 웹사이트의 운영상 발생되어지는 문제점들이 마케팅이나 그래픽 디자인, 탐색, 정보 디자인 자체의 잘못도 있지만, 근본적으로는 분석과 기획, 설계 단계의 결합에서 비롯되는 경우가 많으며, 이러한 경우, 보다 심각한 장애를 일으킬 소지가 높기 때문이다. 국내의 경우도 인터넷 비즈니스 관련업체 대부분이 관리 문제를 뒷전으로 미루는 현 상황에서 '선 구축, 후 유지보수'라는 과정을 되풀이하고 있으며, 이는 개발비용을 훨씬 더 초과하는 유지보수 비용과 기업이 서비스에 대한 관리능력이 라는 심각한 문제점을 야기시킬 수 있다.

2 웹사이트 개발을 위한 개발방법론

2.1 최근동향

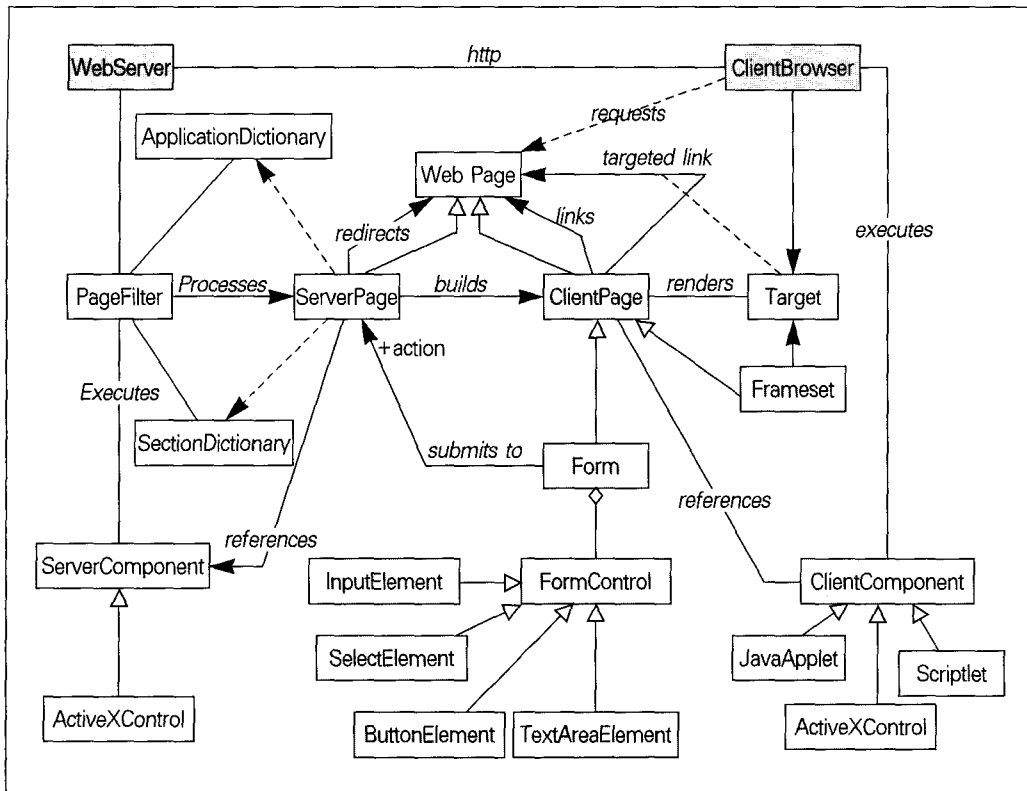
웹사이트 개발 과정에 있어 관리의 대상이 되는 범위는 단순히 '웹페이지 자체를 얼마나 잘 구성할 것인가'의 수준에서부터 웹사이트 구축을 위한 기술적인 요소, 웹사이트를 개발하려는 조직의 운영, 그리고 서비스하는 단계까지를 관리의 범위로 두는 개발방법론까지 그 범위의 폭이 광범위하다. 웹사이트 관리의 중점대상 역시 웹페이지 레이아웃에 초점을 두는 그래픽 디자인의 측면부터 시작하여 콘텐츠의 수집과 분류, 페이지로의 매핑과 관련된 콘텐츠 설계로 범위를 확대해 나갈 수 있으며, 웹 애플리케이션을 개발, 운영하는 기술적인 측면에 보다 초점을 둘 수 있다. 웹 페이지 자체를 구성, 관리하는 측면에 관해서는 Web Review사의 'The Navigation and Usability Guide'가 상당히

유용한 정보를 제공하고 있으며(Muehlbauer 1998), 기획, 디자인, 개발, 운영 및 서비스로 이어지는 웹사이트 개발 프로세스 전 과정에 걸친 보다 자세한 내용은 IBM사의 'IBM Ease of Use Web Design Guidelines'를 통해 살펴볼 수 있다(GVU's WWW Survey Team 1998).

웹사이트 개발을 보다 공학적인 측면에서 체계적으로 분석, 설계할 수 있는 방법론 가운데 객체지향 개발방법론인 UML(Unified Modeling Language)을 이용해서 웹 애플리케이션을 디자인하는 과정은 Rational Rose사의 'Modeling Web Application Design with UML'을 통해 보다 구체적인 개발절차를 확인해볼 수 있다(Conallen 1998). 웹 개발자는 <그림 1>과 같은 UML모델링을 통하여 마치 뛰어난 건축가가 잘 디자인된 설계도를 바탕으로 고층빌딩을 만들어 가듯 웹 애플리케이션 개발과정을 단계별로 진행시켜 나갈 수 있는데, 이는 현재 점점 더 복잡해지고 포털사이트화 되어 가는 웹사이트 개발 상황에 있어 상당히 유용한 도구로 활용되어질 수 있다.

2.2 개발방법론

최근의 웹사이트개발은 점점 복잡화, 대형화되어 발전해 오면서 더 이상 과거의 개발방식인 소위 無방법론, '선 구축, 후 유지보수' 식으로는 대응하기가 매우 어려워졌다. 이는 큰 규모기관의 정보센터와 기업의 정보시스템개발 시의 상황이 매우 유사한데, 이들 기관의 경우에는 업무의 효율을 높이는 경영정보시스템(Management Information System)에서 정보시스템을 이용한 경쟁우위 창출의 개념인 전략적 정보시스템



〈그림 1〉 웹 어플리케이션 설계 모형

(Strategic Information System)으로 요구사항의 수준이 상승되면서 시스템 전략의 계획에서부터 철저한 사용자의 요구분석과 설계를 통해 자동화 툴(tool)에 의한 개발 공정에 이르기까지 그 생산 방법의 자동화 및 체계화가 필요하게 되었다. 이를 해결하기 위한 효과적인 수단 가운데 하나가 개발방법론의 적용이며, 개발방법론은 어떠한 시스템을 개발하기 위한 작업방법이나, 절차, 산출물, 기법 등을 논리적으로 정리해 놓은 체계라고 정의할 수 있다. 마치 빌딩을 지을 때 설계도에 따라 체계적으로 건축이 이루어지듯이 개발자들은 개발방법론을 이해하고 참조하면서 시스템의 계획, 분석, 설계, 구현, 운영의 시스템개발 수명주기(System Development Life Cycle)를

따라 시스템 개발을 수행하게 된다. 개발방법론은 바로 시스템 개발의 이론적 기반이라고 할 수 있다. 방법론의 내용을 구성하는 주요 구성요소로는 다음과 같은 것들이 있다.

- 작업절차 - 프로젝트 수행시 이루어지는 작업단계의 체계
- 작업방법 - 각 단계별로 해야할 일들의 구체적인 설명
- 산출물 - 단계별로 만들어야 할 산출물의 목록과 작성방법, 또는 양식 등
- 기법 - 각 단계별로 작업 수행시에 소요되는 기술 또는 기법의 설명
- 관리 - 프로젝트 관리자의 입장에서 수행해야 할 작업

- 툴(tool) - 상황에 따라 필요한 툴과 그 적용 방법

하지만 개발방법론은 단순히 시스템을 개발하는 개발자만을 위한 도구는 아니다. 개발방법론의 단계별 개발관련 산출물은 업무 현황을 단순화시켜 사업 전반을 총체적으로 이해할 수 있는 청사진 역할을 수행함으로써 운영자에게는 현실태와 문제점 파악을 용이하게 하여 앞으로의 운영개선방향을 제시하게 되며, 신규직원에게는 사업전반에 대해 보다 쉽게 이해할 수 있는 기회를 제공함으로써 신속한 업무응답이 가능토록 해주게 된다. 개발 방법론을 실제 업무에 적용함으로써 얻게되는 이점은 다음과 같다.

첫째, 개발방법론은 현실을 단순화시켜 사업 전반을 총체적으로 이해할 수 있는 청사진을 제시한다.

둘째, 개발방법론은 사업 전반을 현재 또는 앞으로 원하는 모습으로 가시화되도록 도와 준다.

셋째, 개발방법론은 사업의 구조와 안팎에서

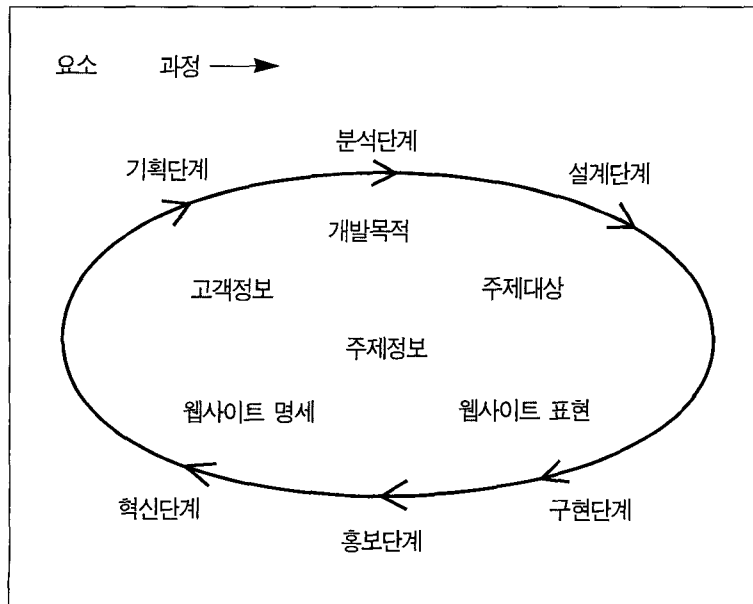
이루어지는 행위를 명세화 할 수 있다.

넷째, 개발방법론은 잠재된 문제점과 예상치 못했던 문제점까지도 인지할 수 있도록 해 준다.

다섯째, 개발방법론은 우리가 결정한 사항들을 문서화한다.

3 웹사이트 개발 절차

보통 웹사이트 개발은 처음 기획단계에서 시작하여 웹사이트를 디자인하고 구현하며 서비스하는 단계에 이르기까지 일련의 개발절차를 따르게 된다. 국내의 경우 이러한 일련의 절차는 대략 3개월 이내에 끝내는 경우가 대부분이고 큰 규모 사이트의 경우는 6개월 정도의 기간이 소요되는데 비하여, 외국의 기업사이트들은 거의 8개월에서 1년에 걸쳐 각 단계별로 충분한 검토와 계획 하에 개발을 진행시키고 있다. 충분한 개발 능력을 갖



〈그림 2〉 웹사이트 개발 절차

준 개발 조직일수록 처음 기획단계에서 소모되어 지는 시간과 노력이 가장 큰 비중을 차지하게 되는데, 기획단계에서 앞으로 소요될 인력과 개발계획에 대한 충분한 검토가 이루어지지 못할 수록 전체 개발 기간의 지연과 그에 따른 초과비용의 지출이 점점 더 늘어날 수 있기 때문이다.

웹사이트 개발절차는 목적과 방법에 있어 약간씩의 차이를 가질 수 있으나, 웹사이트 개발 컨설팅 업체인 December Communication의 사장이면서 웹사이트 개발전문가인 John December의 일반적인 웹사이트 개발 절차를 살펴보면 다음과 같다(December 2000). December는 소프트웨어 개발자로서의 경험과 다년 간의 웹사이트 개발관리 경험을 바탕으로 웹사이트 개발절차를 자신만의 고유한 웹 개발방법론으로 정립하였으며, 일반적인 웹사이트 개발 가이드라인에 비하여 보다 체계적인 지식을 제공하고 있다. December의 웹 개발절차는 <그림 2>와 같이 총 6단계로 되어있으며, 각각의 단계별 절차는 다음과 같다.

- ① 기획(Planning) : 대상고객, 개발 목적 그리고 웹 개발 정책을 명확히 한다.
- ② 분석(Analysis) : 표현적 측면, 기술적 측면, 내용적 측면에서 웹사이트가 잘 구성되었는지를 분석하게 된다.
- ③ 설계(Design) : 각 웹페이지의 구성과 웹 페이지들로 구성된 전체사이트의 구조를 디자인한다.
- ④ 구현(Implementation) : HTML을 이용하여 각 웹페이지를 작성하고 전체사이트의 구조를 만들어간다.
- ⑤ 홍보(Promotion) : 기존의 이용자와 잠재적인 사용자그룹을 대상으로 홍보에 들어간다.
- ⑥ 혁신(Innovation) : 웹사이트를 구성하고 있는 시스템과 콘텐츠를 대상으로 계속적으로 유

지보수가 진행된다.

3.1 기획단계

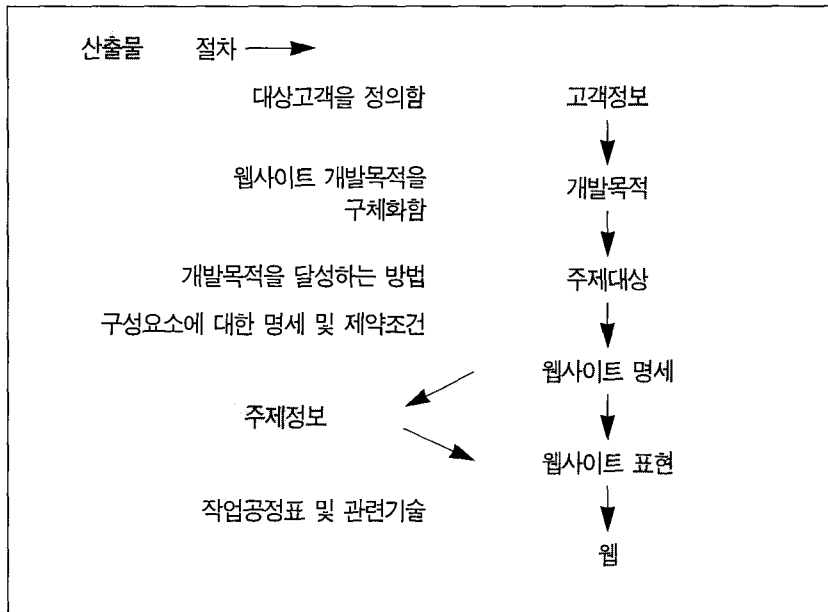
기획단계에서는 웹사이트를 구축하고 운영하는 데 있어 필요로 하는 여러 가지 요소들을 고려하게 된다. 만약 웹 페이지를 디자인하는 단계에 있어 단순히 HTML로 웹 페이지를 구성하는 것이 아니라 CGI프로그래밍을 사용할 예정이라면, CGI프로그래밍을 할 수 있는 전문인력을 확보할 것을 미리 계획해야 한다. 기획단계에서 이루어질 수 있는 일련의 작업들은 다음과 같다.

- 웹사이트 개발에 앞서 예상되어지는 개발 기간 및 비용 산출
- 웹사이트 개발에 필요한 기술 및 정보의 획득에 대한 고려
- 콘텐츠를 수집하고 잘 조직화할 수 있는 방법의 정형화

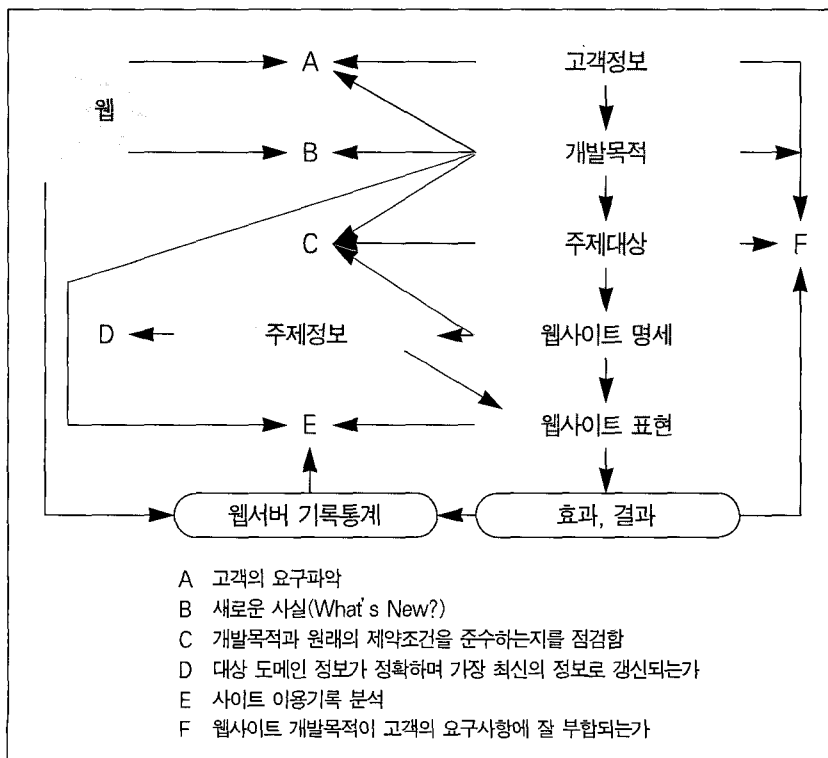
3.2 분석단계

분석단계에서는 웹사이트를 구축하고 운영하는 데 있어 필요한 정보들을 수집하여 비교하는 단계이다. 분석자는 다음과 같은 3가지 측면에서 웹사이트 구축과 관련된 사항을 점검하게 된다.

- ① 수사적(rhetorical) 측면 : 웹사이트 개발과정이 서비스의 대상이 되는 특정 사용자 그룹의 특성을 제대로 잘 반영하고 있는가
- ② 기술적 측면 : 웹사이트 개발과정이 최근 정보기술을 적절히 잘 활용하고 있는가
- ③ 의미적 측면 : 앞으로 서비스하고자하는 주제와 관련되어 수집된 정보가 정말 정확하고 유용한가



〈그림 3〉 기획 단계



〈그림 4〉 분석 단계

3.3. 설계단계

설계단계에서 웹디자이너는 각 웹페이지의 구성과 웹페이지들로 구성된 전체사이트의 구조를 설계한다. 웹페이지의 구성과 웹페이지들로 구성된 전체사이트의 구조를 설계할 때에는 미적인 요소와 정보전달의 효율성측면을 고려해야 하며, 특히 다음단계인 구현단계에서 개발되어지는 웹 어플리케이션과도 상당히 밀접한 관련성을 가지게 되므로 구현단계에서 기술적인 측면을 고려한 설계가 이루어져야 한다. 설계단계에서 고려되어야 할 사항들은 다음과 같다.

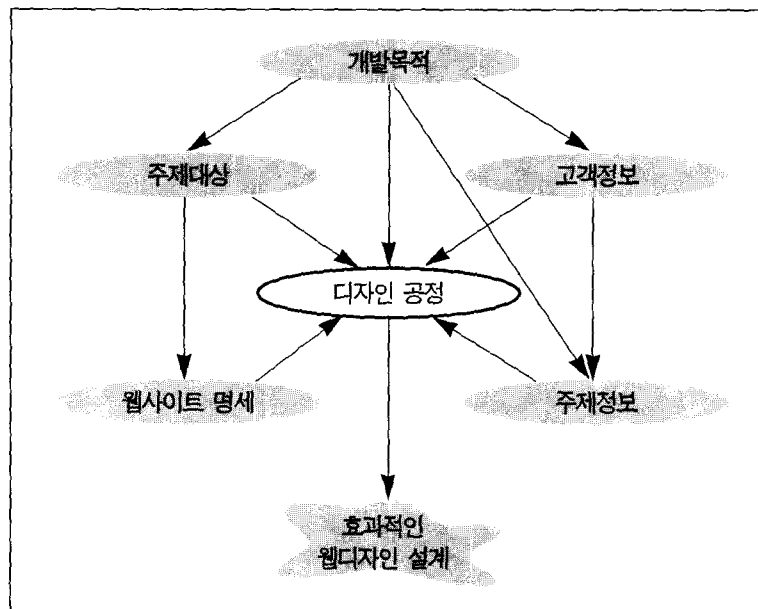
- 웹페이지들로 구성된 전체사이트 구조의 일관성 유지
- 콘텐츠 배치 및 웹페이지 구성
- 실제 이용자의 웹서핑 방식을 고려한 웹사이트의 정보 계층구조설계

3.4 구현단계

구현단계에서는 이전단계인 설계단계에서 만들어진 디자인을 토대로 실제 웹사이트를 구축하게 된다. 웹 개발자는 실제 HTML을 작성하거나 관련 웹 어플리케이션 프로그램을 개발하게 되는데 이러한 구현과정은 소프트웨어 개발과정과 상당히 유사하다. 구현단계에서 고려해야 할 사항들은 다음과 같다.

- 보다 효과적으로 개발작업을 진행할 수 있는 개발툴의 활용
- 추후 확장 가능한 자료구조의 설계
- 다양한 브라우저를 지원할 수 있는 형태로 개발
- 전체사이트 구조의 일관성 유지

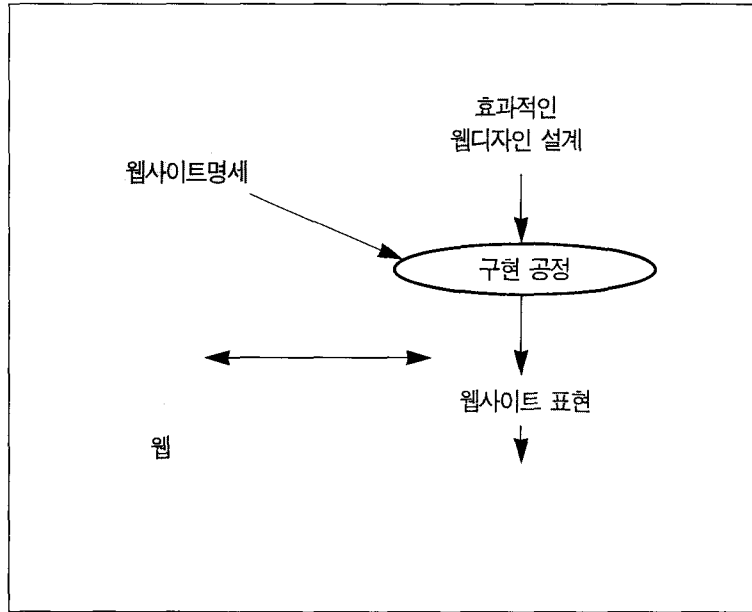
3.5 홍보단계



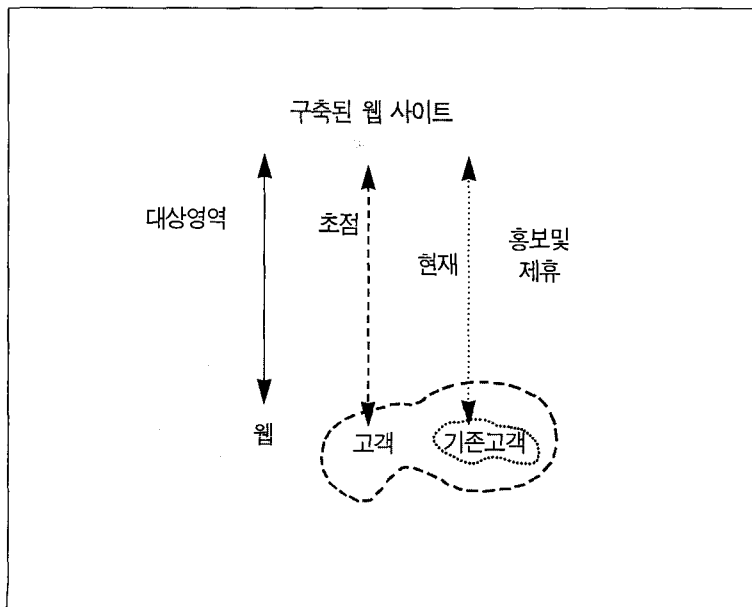
〈그림 5〉 설계단계

홍보단계에서는 개발되어진 웹사이트를 기존의 광고매체를 이용하거나 또 다른 웹사이트를 통해

널리 알리는 단계이다. 보다 적극적인 홍보를 위해서는 특별한 광고전략이나 비즈니스 모델을 이



<그림 6> 구현단계



<그림 7> 홍보단계

용할 수 있다. 홍보단계에서는 다음과 같은 사항이 고려되어야 한다.

- 온라인 공동체에 규범을 준수하는 범위 내에서 적극적인 홍보활동
- 사전 설문조사처럼 실제 이용자와 접촉을 통한 이용자의 잠재욕구 파악
- 현재 고객뿐 아니라 잠재적인 이용자그룹을 고려한 홍보전략

3.6 혁신단계

혁신단계는 기존 소프트웨어개발에 있어 유지보수단계에 해당한다. 웹 개발환경은 소프트웨어 개발환경에 비해 변화의 주기가 짧으며, 웹 개발자는 급변하는 주변환경에 따라 신속하게 웹사이트를 재구성해야하는 임무가 주어진다. 이러한 이유 때문에 웹 사이트를 유지, 혹은 재구성하는 현 단계를 혁신(Innovation)이라는 용어로 표현할 수 있다. 혁신단계에서는 다음과 같은 사항이 고려되어야 한다.

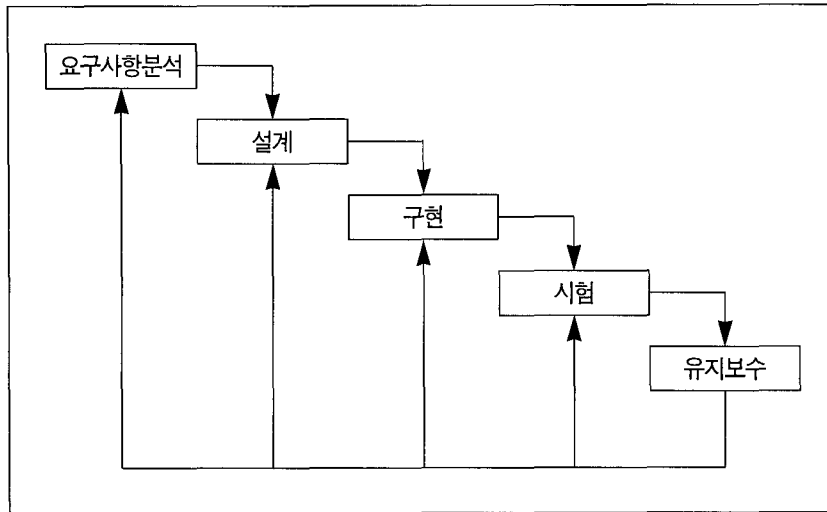
- 이용자의 요구를 반영하기 위한 창조적이며 계속되는 유지보수작업
- 새로운 이용자의 요구를 파악하기 위한 분석을 토대로 웹사이트를 재구성
- 보다 나은 서비스 제공을 위해 지속적으로 신기술 도입

4 웹사이트 공학

웹사이트 개발에 있어서 필요한 표준 개발 방법론이나 프로젝트 관리 등에 있어 기존 소프트웨어 공학의 방법론을 웹사이트 개발에 적용시키고자 하는 일련의 시도를 웹사이트 공학(Web

Site Engineering)이라 정의할 수 있다(David, Dominique and Thomas 1998). 지금까지 웹사이트는 그래픽 디자인이나 정보설계, 사용자 인터페이스 등을 중심으로 개발되어 왔다. 하지만 웹사이트가 점점 복잡해지고 포털 사이트화 되어가는 현 상황에서 웹사이트 개발에 있어서 생산성이나 개발비용 및 개발일정, 품질에 대한 문제점들이 점점 더 부각되고 있다. 그와 아울러 웹사이트 개발에 있어서 품질 보증 활동도 거의 이루어지지 않고 있는 점도 심각한 문제점으로 지적되고 있다. 이러한 상황은 과거 소프트웨어 개발에서 이미 발생되었던 상황으로 1960년대 하드웨어 기술 발전 속도가 소프트웨어 기술 발전 속도보다 빨라서 새로운 소프트웨어에 대한 시장의 요구를 감당하기 힘들어 졌으며, 소프트웨어에 대한 유지보수비용이 전체 비용의 70%를 넘게 되는 문제가 발생하게 되었다. 이러한 문제점들은 점차 인건비 상승 효과와 우수한 소프트웨어의 부족현상으로 악화되어 소프트웨어 생산성 문제로 발전하게 되었으며, 이를 '소프트웨어 위기'라고 한다. 이러한 개발 예산 초과, 개발 일정의 지연, 저조한 생산성, 미흡한 품질 등을 해결하기 위한 노력으로 소프트웨어 개발에 구조적이고 공학적인 접근방법을 시도하여 소프트웨어 공학(Software Engineering)이란 학문으로 이러한 문제들을 연구해 오고 있다(Humphrey 1995; Sommerville 1996; Pressman 1997).

본 연구에서는 소프트웨어 공학의 대표적인 소프트웨어 개발방법론인 폭포수모델, 프로토타입 모델, 나선형모델에 대해 분석해보고, 각각에 대해 웹사이트 개발을 위한 방법론으로서의 활용방안과 실제 방법론으로 활용 시에 발생하게 되는 문제점을 찾아 보고자 한다.



〈그림 8〉 폭포수 모델

4.1 폭포수 모델(Waterfall Model)

가장 오래되고 널리 사용되는 모델로 고전적 라이프사이클 모델이라고도 할 수 있는 방법론으로, 요구사항 분석, 설계, 구현(프로그래밍), 시험 및 유지보수의 순서로 시스템의 개발이 진행된다. 폭포수 모델은 개념 정립에서 구현까지 하향식 접근 방법을 사용하여 높은 추상화 단계에서 낮은 추상화 단계로 옮겨가는 모델로 각 단계가 끝날 때마다 과정의 끝을 알리고 그 다음 단계로 진행하게 된다. 하지만 실제로 소프트웨어 시스템을 개발하다 보면 개발의 각 단계에 겹쳐지는 부분이 나타나게 되며, 각 단계의 진행 과정에서 문제가 발생하여 그 이전 단계로 피드백되는 경우가 발생하게 된다. 〈그림 8〉은 폭포수 모델에서 피드백이 요구되어 순환되는 모습을 나타내고 있다. 각 개발 단계는 약간의 피드백이 이루어진 후 문서화 작업같은 산출물을 남기게 되고, 그 다음 단계로 넘어가는 것이 일반적이다. 폭포수 모델에서 개발 단계는 다음과 같다.

① 요구사항 분석

- 사용자 요구사항을 정의하기 위하여 시스템의 요구사항을 수집한다
- 시스템의 목표를 정하는 과정으로 그 결과물은 요구사항 명세서이다

② 설계

- 설계는 요구사항 분석과정에서 모아진 요구사항을 설계도면에 옮기는 과정이다
- 설계 과정은 물리적(physical) 실현의 첫 단계
- 설계 단계의 결과물은 설계 명세서로 나타남

③ 구현

- 시스템의 기능이 수행 가능한 모습으로 나타남
- 구현은 프로그래밍 또는 코딩이라고 부른다
- 프로그래밍의 결과는 컴퓨터 프로그램

④ 시험

- 품질보증 활동의 중요한 일부분
- 사용자 요구사항, 설계, 구현의 전 과정에 대한 최종 점검을 포함
- 시험은 제품의 오류를 발견하고 수정하는 과정
- 최소한의 시간과 비용을 투자해서 최대한의 확

물로 오류를 찾아낼 수 있도록 이루어져야 함

⑤ 유지보수

- 여러 변경 사항에 대해 적응하는 활동이며 변화에 대비하는 과정이다
- 수정 유지보수, 적응 유지보수, 기능추가 유지보수, 관리 유지보수

폭포수 모델은 프로젝트 진행과정을 세분화하여 진행함으로써 관리가 용이하다는 장점을 가진다. 하지만 대부분 경우 개발과정에 있어 피드백이 발생하기 때문에 순차적인 흐름을 따라가는데 어려움이 있다. 특히 웹사이트개발에 있어서는 각 단계가 중복되거나 반복되는 경우가 많기 때문에 폭포수 모델은 별 도움이 못된다. 간단히 말해서 폭포수 모델은 웹의 빠른 변화와 유동성(flexibility)을 감내하기에는 적합하지 못한 모델인 것이다. 또한 고객이 원하는 요구사항을 초기에 구체적으로 기술하기도 어렵다. 폭포수 모델에서 개발자는 다음 단계에 대한 계획을 확실하게 세울수록 좋다. 하지만 빠르게 변화하고 있는 최근의 개발환경에서 앞으로의 계획을 구체적으로 한다는 것 자체는 불확실하기 때문에 어려운 작업이 된다. 특히 웹사이트 개발에서 기존의 웹사이트를 개발한 경험이 없는 개발자는 다음 단계에 대한 완벽한 계획을 세우는 것이 매우 어렵다. 그럼에도 불구하고 폭포수 모델은 웹사이트 개발에 있어 가장 널리 쓰이고 있는데 각 단계별 절차를 이해하기 쉽고 쉽게 작업진행 상황을 파악할 수 있기 때문이다. 폭포수 모델은 규모가 작고 구체적인 개발목표가 정립되었을 경우의 웹사이트를 구축할 때 효과적일 수 있다.

4.2 프로토타입 모델(Prototype Model)

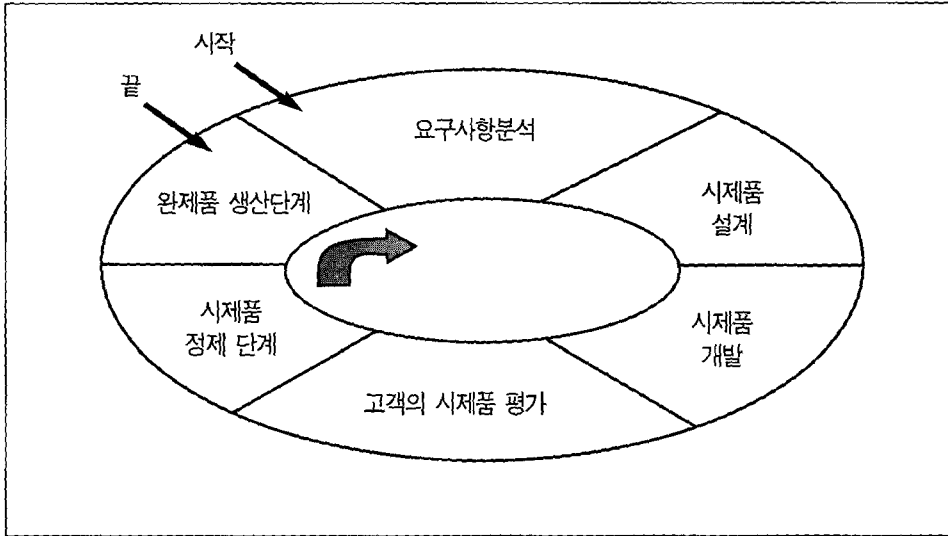
소프트웨어나 웹사이트 개발에 있어 고객이 목

표를 정의하였으나 고객의 요구를 어떻게 만족시킬 수 있을지 모르는 경우가 자주 있다. 특히 이용자 자신도 자신이 원하는 것이 무엇인지 구체적으로 모르거나 그들의 요구가 어떻게 변경될 지 잘 알지 못하는 때가 허다한 실정이다. 이러한 이유 때문에 종종 개발자들은 고객의 요구를 불완전하게 이해하고 있는 상황에서 개발을 진행시켜야 한다. 이런 문제점을 해결하기 위한 방법으로 간단한 시제품(prototype)을 만들어 사용자의 요구를 수용하여 시스템을 보완해 가는 방법이 프로토타입 모델이다. 프로토타입 모델은 폭포수 모델의 단점을 보완하기 위해 점진적으로 시스템을 개발하여 나가는 접근 방법을 취하고 있다.

프로토타입 모델은 실제 사용자로부터 피드백을 시스템 개발 초기에 얻어낼 수 있다는 장점을 가지고 있다. 특히 시제품을 통해 이전에 밝혀지지 않았던 사용자의 요구사항을 구체적으로 규명할 수 있으며, 프로젝트 초기에 요구사항이 확실치 않거나 모든 요구사항을 미리 뽑아낼 수 없는 불안정한 상황일 때 보다 유용한 방법이다. 개발팀은 사용자 및 고객의 대화를 통하여 전반적인 기능을 파악하고, 우선 간단히 설계를 한 후 시제품을 만들어 사용자에게 보여주게 된다. 사용자는 시제품에 대하여 평가하고 그 결과는 시제품을 향상 시키거나 완제품을 만들어 가는데 반영하게 된다. 사용자로부터 피드백을 얻어낸 후 시제품을 버리는 경우도 있고, 원하는 시스템의 기능 중 중요한 부분만 구현하여 피드백을 얻은 후 계속 발전시켜 완제품을 만들어 낼 수도 있다. 프로토타입 모델에서 개발 단계는 다음과 같다.

① 요구사항 분석 단계

- 이 과정은 폭포수 모델의 요구사항 분석단계와 유사하다
- 고객으로부터 받은 일부의 요구 사항만 정의



〈그림 9〉 프로토타입 모델

- 하고, 완전치 않은 요구사항에 대하여 윤곽을 잡아 놓는다
- 추가적인 정의가 필요한 부분은 시제품이 개발된 후 계속 정제해 나간다.
- ② 시제품 설계 단계
- 프로토타입에 대한 설계를 한다.
 - 사용자들이 볼 수 있는 면에 초점을 맞춘다
 - 시제품 개발의 목표가 확립되고 시제품에 포함될 시스템의 기능들이 골라진다.
 - 시제품에 포함되는 것과 시제품에서 배제되어야 하는 것이 무엇인지 규명하는 것이 중요하다
- ③ 시제품 개발 단계
- 일반적으로 성능, 다른 시스템과의 인터페이스 등에 대한 것은 판단하기 어려워 중요하게 다루어지지 않는다.
 - 오류나 예외처리를 해결하는 부분은 무시되거나 기초 수준 정도로 구현한다
 - 시제품의 신뢰도와 프로그램 품질 수준은 떨어진다.
- ④ 고객의 시제품 평가 단계
- 프로토타입 모델의 가장 중요한 단계라 할 수 있다.
 - 시제품은 고객에 의해 평가되고, 개발될 소프트웨어의 요구사항을 구체적으로 정제하기 위해 사용된다
 - 이를 통해 요구사항의 오류를 발견하고 규명할 수 있게 되며, 추가되어야 하는 요구사항을 찾아 낼 수 있다.
- ⑤ 시제품 정제 단계
- 사용자가 원하는 것을 만족시키기 위해 시제품에 대한 조율이 필요하다
 - 시제품이 어떻게 고쳐져야 하는지 결정하고 다음 단계의 시제품이 빠르게 만들어 질 수 있도록 한다.
 - 이 시제품은 다시 고객에게 평가되는 순환을

하게 되면 고객이 요구사항에 대하여 만족할 때까지 계속하여 향상된 시제품을 개발하게 된다

⑥ 완제품 생산 단계

- 이 단계의 목표는 원하는 시스템을 개발하는 것이다
- 만약 프로토타입을 버리고 새 시스템을 개발해야 한다면, 폭포수 모델의 생명 주기를 따라 개발하는 것이 효과적이다.

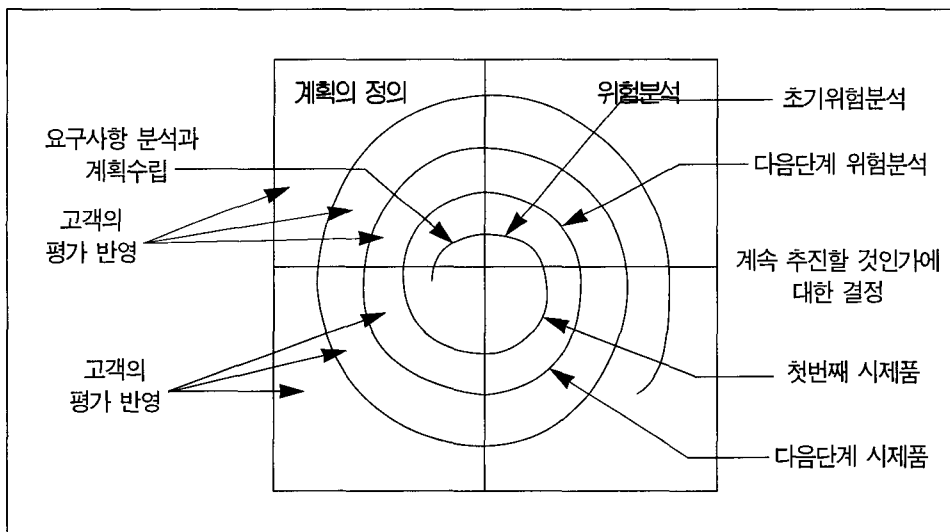
웹사이트구축에서 프로토타입 모델은 사용자의 요구를 적극 수용해야 할 경우 도입하게 된다. 폭포수모형을 도입한 사이트보다 훨씬 훌륭한 웹사이트를 구축할 수 있으며, 개발시부터 사용자의 요구를 철저히 분석하게 되므로 실패할 확률이 적다. 그러나 시제품을 대상으로 시험, 평가를 수행할 평가자를 모집, 관리해야하는 어려움이 따르며, 사용자 그룹에 대한 정의가 명확하지 않을 경우에는 개발이 사실상 불가능해진다. 특히 일반이용자를 대상으로 제한 없이 시제품평가를 시

행할 경우, 잠재적인 고객을 떠나버리게 만드는 최악의 상황을 유발할 수 있다. 시제품은 어디까지나 조직 내부에 시스템 평가 목적으로 사용되어야 한다. 대충 빨리 웹사이트를 만들어 시제품을 가지고 서비스를 시작하고 고객의 불만사항을 점차 해소해 나가겠다는 생각은 웹사이트개발 자체를 실패로 이끌어 가는 가장 치명적인 요인이 될 수 있음을 명심해야 한다.

4.3 나선형 모델(Spiral Model)

나선형 모델은 폭포수 모델과 프로토타입 모델의 장점에 새로운 요소인 위험 분석(risk analysis)을 추가하여 만든 모델로, 시스템을 개발하면서 생기는 위험을 관리하고 최소화하려는 것이 이 모델의 주된 목적이다. 나선형 모델은 나선을 따라 돌면서 점진적으로 완벽한 시스템을 개발하게 되며, 각 나선은 다음의 4단계로 나뉘어져 있다.

① 계획 및 정의 단계



<그림 10> 나선형 모델

- 요구사항을 모으고 프로젝트 계획을 수립한다.
- 나선형 싸이클의 시작은 성능, 기능을 비롯한 시스템의 목표를 규명하는 것에서 시작
- 시스템의 목표와 제약조건에 대한 차선책이 평가, 고려될 수 있다.
- 이러한 평가과정은 프로젝트 위험의 원인을 규명하는데 효과적으로 사용된다.

② 위험 분석 단계

- 여기서는 초기 요구사항에 근거하여 위험이 규명된다.
- 정보를 찾아내는 활동을 통하여 불확실성과 위험을 줄여나갈 수 있다.
- 프로젝트를 '계속 진행할 것인지', 또는 '중단할 것인지'를 결정하는 작업이 이루어진다.

③ 개발단계

- 이 과정은 위험에 대한 평가가 있는 다음 이루어진다.
- 이 단계에서는 '어떠한 패러다임이 적용되며 시스템 개발이 이루어 질 것인가' 하는 개발 모델을 결정한다.
- 이 단계는 시제품을 개발하거나 최종 제품을 만드는 과정이라 볼 수 있다.

④ 고객 평가 단계

- 이는 앞의 결과를 사용자가 평가하는 과정이다.
- 고객에 의해 시스템에 대한 평가가 이루어지고 고객은 시스템의 수정을 요구하기도 한다.

나선형 모델은 비용이 많이 들고 시간이 오래 걸리는 관계로 대규모의 개발이 아닌 일반적인 웹사이트 개발에 있어서는 부적합한 면이 있다. 또한 모델 자체가 앞의 두 모델보다 더 복잡하여, 분석자가 나선형모델의 적용경험이 충분치 못한 경우 프로젝트 관리 자체를 어렵게 만들 가능성이 많다. 나선형 모델은 대규모의 웹사이트 신규개발에 적합한

모델로 이미 구축되어 있는 사이트를 보수하려 하거나 효용성이 없는 사이트를 구축하기 위해서 무작정 나선형 모델을 적용한다면 예산과 시간낭비라는 큰 문제에 부딪히게 될 것이다.

5 결론 및 제언

지금까지 살펴본 바와 마찬가지로 현재 웹사이트 공학 연구자들은 주로 소프트웨어 공학적인 측면에서 웹사이트를 분석, 모델링하고 있는데, 과연 웹사이트를 소프트웨어의 일종으로 볼 수 있을 것인가는 보다 깊이 생각해볼 필요가 있다.

초창기에 웹사이트는 단순히 출판물의 출판 대상이 서지의 형태에서 웹이라는 전자매체로 바뀌었을 뿐이다. 다시 말해 사용자에게 일방적으로 정보를 제공하는 서지 정보의 형태에서 전자매체 형태로 온라인 상에 제공하는 수준에 그쳤을 뿐이다. 하지만 웹사이트 기술이 발전하고 방문자의 요구사항이 다양해지면서 웹사이트는 일방적인 정보전달의 수단보다는 이용자와 서비스 제공자가 서로 교류하는 형태로 발전해가고 있다. 서지 정보의 형태에서 상호작용을 중시하는 소프트웨어로 발전해가는 웹사이트는 기존 소프트웨어와의 구분도 모호해지게 되었으며, 이러한 관점에서는 웹사이트도 소프트웨어의 일종이므로 소프트웨어 공학의 교훈을 얻을 수 있을 것이다.

하지만 웹사이트를 실제로 제작하고 있는 개발자들과 대부분의 이용자들은 "웹사이트를 한번도 소프트웨어라고 생각해 본 적이 없다"고 말한다. 웹사이트는 복잡적이어서 단지 소프트웨어라고만 보기는 어렵다. 이러한 측면에서 살펴볼 때 웹사이트 구축관련 소프트웨어는 웹사이트를 이루는 요소 중 일부일 뿐이다. 이는 인터넷 관련 기술을

판매하는 몇몇 회사를 제외하고 잡지, 신문, 방송, 디렉토리 서비스와 같은 대부분 형태의 서비스에서 공통적 현상을 보인다. 인터넷 서점인 아마존에서 웹사이트 구축관련 소프트웨어는 “책을 인터넷에서 판매한다”는 기능을 수행하기 위한 보조도구이지 그 자체가 주가 되지는 않는다. 이러한 관점에서 살펴보면 소프트웨어 개발 프로세스 모델 측면에서 웹사이트 공학의 개발방법론은 기술, 공학적 접근 방식에 근거를 둔 시스템 조직 내에서의 기술적인 변화과정을 관리하고 새로운 시스템을 개발하기 위한 방법을 위주로 해결책을 제시해줄 뿐이다. 따라서 보다 효과적인 업무분석과 개발관리 방법을 제시하기 위해서는 소프트웨어 개발 위주의 측면이 아니라 실제 서비스되는 콘텐츠와 실제 업무를 수행하는 조직과 조직 내 사람, 그리고 조직간에 활동을 포괄하는 전체 시스템을 분석, 점검할 수 있는 시스템적 접근방식(Underwood 1996; Meadows 1989)에서의 개발방법론을 필요로 하게 된다.

새로운 웹사이트 개발에 있어 또 다른 문제점으로는 급속도로 변해가고 있는 인터넷 상에서 웹사이트의 개발은 운영자와 개발자 모두에게 큰 부담으로 작용하게 된다는 점이다. 따라서 실제 업무를 수행하고 있는 운영자가 개발방법론을 익혀 실제 업무에 적용하기까지 상당한 기간과 노력을 필요로 한다면 웹사이트 개발 책임자는 개발방법론을 통해 얻을 수 있는 여러 가지 이점에도 불구하고 기존의 방식을 고수하게 된다. 지금까지 언급되어진 내용을 토대로 웹사이트 구축을 위한 웹사이트 개발방법론이 갖추어야 할 요건을 결론적으로 정리하여 제시하면 다음과 같다.

① 실용성

웹사이트 개발방법론은 실제 업무에 적용 가능해야 하며, 새로운 웹사이트 개발방법론의 개발

은 연구자 위주가 아닌 실제 업무관련 실무자의 노하우를 근간으로 해야 한다.

② 간편성

비전문가라도 실제 업무에 적용 가능하도록 복잡하지 않으면서도 효과적인 방법이어야 한다. 기존 소프트웨어 공학 개발방법론의 경우, 비전문가는 실제 업무에 개발방법론을 적용하기 위해서는 별도의 교육과정이 필요하며, 효과적인 분석이 가능해질 때까지 오랜 기간 동안의 연습을 필요로 하였다. 따라서 비전문가라도 최소의 노력으로 최단 시간 내에 실제 업무에 적용 가능한 새로운 웹사이트 개발방법론을 필요로 하게 된다.

③ 유연성

인터넷상에서 실제 웹사이트를 방문하는 사용자 계층의 취향은 언제 어떠한 형태로 변할 지 예측하기가 상당히 어렵다. 따라서 웹사이트의 구조와 서비스 방식은 하나의 상품명과 일정기간의 생명주기를 가진 소프트웨어와는 달리 급변하는 네티즌의 취향에 발맞춰 구조적으로나 내용적으로 계속 변해야만 하는 상황에 처하게 된다. 이러한 상황에서 웹사이트 분석자는 웹사이트 개발방법론을 이용하여 새로운 변화에 발맞춰 신속한 재모델링이 가능해야 한다.

④ 콘텐츠 기반 vs. 데이터 기반

웹사이트는 콘텐츠 기반이고 소프트웨어는 데이터 기반이다. 웹사이트 개발에 있어 데이터는 콘텐츠의 일부일 뿐이다. 웹 서핑 한다는 것은 인터넷이란 가상 세계에서 정보를 찾아다니는 행위인 것이다. 따라서 기존의 데이터 기반 소프트웨어 개발방법론과 콘텐츠 기반 웹사이트 개발방법론은 본질적으로 큰 차이점을 가지게 된다.

현재 조직적이고 체계적인 웹사이트 개발과 운영이라는 이론적 개념 자체가 거의 없는 국내 웹사이트 개발상황하에서 국내여건에 맞는 웹사이트

개발방법론 개발은 무한경쟁에 가까운 인터넷 비즈니스 환경 하에서 국내 인터넷비즈니스 업계가 생존하기 위한 가장 시급한 과제일 수 있다. 이러한 국내 여건 하에서 새로운 한국형 웹사이트 개발방법론 개발은 기존 정보학 연구자들의 임무임을 통감하고, 실제 인터넷 비즈니스를 수행하고 있는 기관 및 업체 등의 개발노하우를 바탕으로

시스템적 접근방식에서 효과적인 웹사이트를 개발, 운영하기 위한 개발방법론을 새로이 연구개발해야 한다. 인터넷 비즈니스 분야의 전문가가 절대적으로 부족한 국내 여건 하에서 앞으로 꾸준하게 웹사이트 개발에 관한 다양한 연구가 진행되어 국내 인터넷비즈니스 학계 및 업계의 대외경쟁력에 진정한 기반이 되어야 할 것이다.

참 고 문 헌

- 『가트너그룹의 IT조명 - 프로젝트, '방법론' 만으로 장담 못해』. 1999. IT Business. 9월 1일. <<http://www.itbiz.co.kr/news/cio/index.htm>>.
- 『기업 포털: 어떻게 관리할 것인가-관리 기능 빠진 포털은 '속빈 강정'』. 1999. IT Business. 6월 1일. <<http://www.itbiz.co.kr/news/cio/1999060105.htm>>.
- Muehlbauer, Jen. 1998. "The Navigation and Usability Guide." Web Review. <<http://webreview.com/wr/pub/98/05/15/thing>>.
- GVU's WWW Survey Team, 1998. "IBM Ease of Use Web Design Guidelines." <http://www-3.ibm.com/ibm/easy/eou_ext.nsf/publish/572>.
- Conallen, Jim. 1998. "Modeling Web Application Design with UML." <http://www.rational.com/uml/resources/whitepapers/dynamic.jtmpl?doc_key=100462>
- December, John. 2000. "Web Development Methodology Overview." <<http://www.december.com/present/webweave.html>>.
- Jones, David L., Dominique C. Cutts and Thomas A. Powell. 1998. *Web Site Engineering: Beyond Web Page Design*. Prentice Hall.
- Humphrey, Watts S. 1995. *Discipline for Software Engineering* (Sei Series in Software Engineering). Addison-Wesley.
- Sommerville, Ian. 1996. *Software Engineering*. Addison-Wesley.
- Pressman, Roger S. 1997. *Software Engineering : a Practitioner's Approach*. McGraw-Hill.
- Meadows, J. 1989. "Evaluating the effectiveness of information centres and services." AGARD(Advisory for Aerospace Research & Development) lecture series. NATO : North Atlantic Treaty Organization.
- Underwood, P. 1996. *Soft System Analysis and the Management of Libraries Information Services and Resources*. London: Library Association.