

인터페이스 구조에 따른 프로토타입의 반응형 웹

장승영*

Responsive Web of Prototype based on Implementation Structure

Seung-Young Jang *

요 약

본 논문은 인터페이스 구조를 근거로 프로토타입의 웹 사이트 설계 방식에 대한 반응형 웹의 설계에 필요한 기술들을 두고 서술하게 되었다. 반응형 설계는 프로토타입의 요구사항을 인터페이스 기반사항에 근거로 분석하여 개발과정에 적용하는 프로세스를 의미한다. 개발에 있어서 우선원칙에 의한 설계방식을 수립하고 디자인공식에 의한 유연성과 가변성을 설정하여 사용자중심의 제1원칙과 제2원칙으로 나누어 인터페이스 구조로 정립해 나갔다. 그리고 압축기술을 논의를 통해 비동기적 스크립트를 결합방식의 방법을 모색하여 웹의 프로토타입에 맞게 확장시킬 수 있는 방법을 찾게 되었다. 오늘날 웹 사이트나 웹 애플리케이션에 대한 필요성은 장치에 관계없이 적용하고 접근할 수 있어야 한다. 이러한 해결방안으로 반응형 웹 사이트 설계방식에서 찾게 되었으며 반응형 웹 사이트 설계방식에 대한 인터페이스 구조와 이를 근거로 한 프로토타입의 웹 사이트 설계방식에 대한 중요기법들을 마련하게 되었다.

ABSTRACT

The purpose of this study is to describe the technologies necessary to design responsive web with the use of the responsive website based on implementation structure as a prototype. Responsive design means the process of analyzing prototype requirements on the basis of implementation and applying the analysis results to development. The design method based on the principle of priority in development was established, and then flexibility and variability through design formula were set up. After that, the user-oriented primary principle and secondary principle were drawn to establish implementation structure. In the way of discussing compression technologies, a method of combination was searched for, and then a method of asynchronous script scalability in line with a web prototype was found. Today, a website or a web application needs to be adopted and accessed regardless of device types. As a solution, a responsive website design method was found. Important technologies for the implementation structure for the responsive website design method and its relevant prototype website were came up with.

키워드

Smart Web, UI, Responsive Web
스마트 웹, UI, 반응형 웹

1. 서 론

1.1 연구필요성

스마트 폰의 보급은 전 세계인구 10명중 4명이 사

용하며 한국은 전체인구의 85%의 이상 스마트 폰을 사용하고 있다. “스트래티지애널리틱스(SA)”에 따르면 전 세계인구 대비 스마트 폰 보급률이 4.8%에 이를 것으로 추정하고 있다. 또한 한국은 월평균 스마트

* 전남과학대학교정보전산원
• 접수일 : 2018. 03. 30
• 수정완료일 : 2018. 05. 07
• 게재확정일 : 2018. 06. 15

• Received : Mar. 30, 2018, Revised : May. 07, 2018, Accepted : Jun. 15, 2018
• Corresponding Author : Seung-Young Jang
Dept. Department of Business Information, NAMBU National University,
Email : sychaing@hotmail.com

폰 데이터 사용량이 4.9GB로 세계 4위를 기록한다[1].

스마트 폰의 확대는 데이터 사용 증가에 따른 무선 트래픽의 증가를 단적으로 보여준다. 한 보고서에 따르면 2013년부터 2018년 사이에 유선 트래픽 증가율 보다 무선 트래픽 증가율이 3배 높을 거라는 전망을 내 놓았다[2]. 우리나라는 모바일 연결사용 비중이 2013년 75%에서 2018년 93%로 증가 추이를 전망하고 있다. 모바일 성장세는 단순히 통화의 목적이 아닌 커뮤니케이션을 넘어선 ‘문화 인터페이스(Manovich, 2001)’의 기능을 수행하고 있음을 의미한다. 디지털문화의 의미란 “전통 미디어(책, 영화, 텔레비전 등)와 마찬가지로 문화형식을 접하는 새로운 ‘문화적 플랫폼’으로 창출 해 나가고 있다.”고 정의를 내린다[3]. 문화적 플랫폼의 성장은 사용자측면에서 보다 기업측면에서 고도성장을 이끌어가고 있다. 문화플랫폼은 기업을 비롯한 많은 프로세스조직 내에 다양한 목적에 의해 경영철학과 마케팅전략에 접목하여 문화변화에 변혁을 일으키는 전략으로 추진한다[4]. 디지털 마케팅시대에 있어 쉽고 명확한 인터페이스가 전략적 핵심가치로 전개되고 있다. 대다수의 소비자 혹은 사용자가 다중플랫폼을 사용하고 있는 사실과 종종 다중스크린을 사용하여 모바일 또는 데스크 톱 사이트에 접속해서 기기전반에 걸친 일관된 환경을 제공할 수 있는 명확한 인터페이스전략이 요구된다. 전략적 인터페이스방식에는 조율수준을 높이는 포지셔닝 전략적 조합이 반영된다. 비즈니스모델에 있어서는 웹 콘텐츠의 편의성이 우선시 되어야 하며 제공을 목적으로 하는 사이트 구조에 있어서도 기기에 대한 디바이스의 제약성 없이 재생산, 재배포 등 사용자 편의성이 우선이 될 수 있는 구조적 인터페이스 전략이 필수라 하겠다.

1.2 연구문제

전통적으로 웹 사이트 구성은 고정된 격자 배치로 설계된다. 고정 격자배치의 설계방식은 그래픽 디자이너들에 의해 대중화 되었는데 이들은 웹 콘텐츠를 배치할 수 있는 모듈을 구성하여 고정된 열과 행을 기준으로 표현해 냈다. 이처럼 고정계층을 고정된 메뉴 표현방식으로 적용하다 보니 설계방식에 있어 유연성의 제약원인으로 영향을 미쳐 데스크 톱 브라우징 사용자보다 모바일기기인 스마트 폰 사용자에게서 상대

적으로 열악한 환경문제로 전이하게 된다. 또한 스마트폰, 태블릿, 데스크 톱 등 각기 다른 디바이스에 대한 각자의 화면해상도 본질여부에서 문제를 파악할 수 있다. 다시 말해 특정 장치인 디바이스에 맞게 각기 다른 해상도에 적합한 설계방식으로 짜여진 틀이라도 정보의 다양성을 배제시켜버린 부분적 웹 설계방식을 적용했기 때문에 사용자 경험을 만족시킬 수 있는 단계까지 미칠 수가 없었다.

일련의 문제과정을 통해서 해상도와 화면크기를 미리 결정하는 모든 기기의 웹 사이트를 각기 다른 설계방식으로 일관성 있게 설계하기란 개발자 입장에서 매우 어렵다. 오늘날 웹 사이트나 웹 애플리케이션의 필요성은 장치에 관계없이 적용하고 접근할 수 있어야 한다. 이러한 해결안으로 반응형 웹 사이트 설계방식에서 찾게 된다.

본 논문은 다양한 장치에서 사용자 경험에 영향을 미치지 않은 웹 사이트를 개발하기 위한 목적으로 반응형 웹 사이트 설계방식에 대한 인터페이스 구조와 이를 근거로 한 프로토폴 타입의 웹 사이트 설계방식에 대한 반응형 웹의 개발에 필요한 중요기법들을 논의하고자 한다.

II. 이론적 논의

2.1 반응적 웹 디자인

유연성과 가변성을 위한 웹 사이트를 설계하는 것을 반응형 웹 디자인(Responsive Web Design, RWD)이라고 한다. RWD의 핵심은 웹의 최적화 구성 방법으로 유연성과 적응성을 사용자 중심의 환경을 제공하는데 초점을 맞추어 있음을 주요골자로 이야기 된다[5]. 이는 웹 사이트의 화면크기, 플랫폼 및 방향에 기초하여 사용자의 행동과 환경에 대응할 수 있도록 단순화시킨 인터페이스 설계방식을 의미한다. 또한 기기마다 다른 버전이 생성되는 기존의 설계 접근방식 대신 대응설계 접근방식을 선택함으로써 응답성이 뛰어난 웹 사이트를 구조로 접근성을 최적화시킬 수 있는 구현방식임을 내세우고 있다. 접근성은 결국 사용자 경험을 향상시키고 사용자가 찾고자 하는 콘텐츠를 쉽게 찾을 수 있도록 상호 연관된 방식을 보여 줄 수 있게 되었다는 점에서 주목한다.

2.2 반응적 디자인 요소

반응형 웹 사이트를 설계하기 위해서는 “우선원리” 설계방식을 수립하게 된다.

“Ethan Marcotte”의 발언에 따르면 반응형 설계에서 3대 요소를 Fluid Grid, Media Query, Flexible Media 가변적 기능식으로 정의하였다[6].

첫째, 반응형 설계에서 중요사항은 가변적선언의 우선원칙에 근간을 둔다. 유동적 구조인 Flexible grid 레이아웃 방식은 레이아웃의 비율을 콘텐츠 영역 내에서 가변적 비율로 선언하게 된다. 가변적 선언은 반응형 디자인 공식인 “(target÷context)×100=result(%)” 식에 의해 가변비율로 선언되어 화면값을 적용하게 되는 구조다[7].

둘째, “Flexible Media”는 영상과 비디오를 웹 콘텐츠에 렌더링 할 수 있는 디바이스에 대한 화면영역 설정방식을 뜻한다. CSS(: Cascading Style Sheets)의 “max-widtht” 속성을 사용하여 상대적 비율값을 “100%”로 적용해서 레이아웃과 동일한 방법으로 속성의 상속을 물려받아 미디어를 확장하는데 안정적인 방식을 응용하게 된다[8].

셋째, “Media Query”는 화면의 공간(여백)을 이용할 수 있는 공간적 외양방식의 요소다. Media Query는 World Wide Web(: W3C)에서 2012년 6월 발표한 CSS버전3에서부터 적용되기 시작했다. 공간설계 있어서 비례영역과 비례영역은 화면의 외형적 치수를 CSS속성의 조건식에 설정하여 논리적 표현방식으로 기능들을 표현하게 된다[9].

2.3 반응형 웹 기술

웹의 적용하는 여러 기술들이 있지만 표준이 되는 기술을 HTML (: Hyper Text Markup Language), CSS(: Cascading Style Sheet), JavaScript로 세 가지 주요기술로 축약된다.

접근성 표준인 HTML5의 기준으로 적용하여 새로운 미디어 요소를 제공하고 디자이너와 개발자들에게 안정적인 플랫폼, 풍부한 경험을 동적 그래픽을 구현할 수 있는 환경여건을 마련했다는 점에서 높은 관심을 보였다[7]. 무엇보다도 주요 브라우저, 모든 스마트폰, 태블릿·멀티스크린 장치가 HTML5를 지원할 수 있다는 점에서 호평을 받고 있다.

CSS은 HTML및 JavaScript와 함께 사용하여 웹

사이트의 프런트-엔드(Front-end)를 만들 수 있는 웹의 스타일시트 코딩방식을 취한다[10]. CSS는 색상, 레이아웃, 폰트와 같은 웹 페이지의 표현을 설명하는 언어이며 다양한 디바이스를 위한 웹 페이지의 시각 및 청각적 레이아웃을 지원할 수 있는 웹 접근성 기술들 중 하나다.

JavaScript는 HTML로 웹의 구조를 정의하고 CSS로 웹의 조성을 구현하여 JavaScript로 웹의 인식 틀을 상호작용 기능 역할을 수행한다.

III. 반응형 웹 디자인 구현 기법

2010년부터 웹 개발 분야에는 엄청난 변화가 있어왔다. 2007년 애플이 아이폰을 출시 시점을 기준으로 하여 2007년 이전과 이후로 나뉘어 사용자 중심(User-Centered Design Process, UCDP)의 모바일 사용성 원칙에 대한 새로운 웹 모델의 패러다임을 불러일으켰다[11]. 사용자 중심의 구현방법에는 사용성 원칙의 제1원칙과 제2원칙으로 나뉜다. 간략하게 제1원칙은 크게 보이는 시각적 접근방식을 뜻하며 제2원칙은 보이지 않는 기반시스템 활용방식을 뜻한다.

3.1 제1원칙 콘텐츠 구현방식

웹 사이트의 주요 목적 중 하나는 사용자가 정보를 얻는데에 있다. 정보는 콘텐츠에서 각각의 요소들을 수집하여 하나의 페이지로 구성한다. 콘텐츠의 각각에 요소들은 텍스트와 이미지로 구성되어 사용자에게 정보를 제공하게 된다.

사용자 중심을 위해 제공되는 콘텐츠는 사용성 원칙의 주요제시 방법 중 제1원칙에 속한다. 제1원칙에는 시각, 행동, 편의 위주의 목적과 의도를 명확히 알리는 체계로 구현한다. 체계적 접근방식은 동일한 레이아웃을 여러 콘텐츠 디자인영역으로 구현할 수 없기 때문에 Media Query, Fluid Grid의 가변적 구문 선언방식으로 설계된다.

3.2 제2원칙 웹 성능의 최적화 구현 방식

사용성 제2원칙에는 시스템 반응의 직관적 성능에 대한 웹 성능의 최적화 구현방식을 말한다. 웹 성능의 최적화 구현방식을 위해서는 비동기 스크립트를 사용

하며 시스템의 구조형태를 우선순위 방식으로 적용하여 신속한 정보를 전달하기 위한 목적으로 선언하게 된다.

- 첫째, 압축기술을 이용한 구조화된 선언
- 둘째, 스크립트와 스타일 시트 간에 결합
- 셋째, HTTP 캐시 헤더 정책 설정

3.2.1 구조화 선언 적용 기법

인터넷 속도를 위해 캐시의 압축기술을 사용하여 비동기 스크립트를 적용해야 한다는 점이다. 인터넷 기반시설의 차이 때문에 불필요한 다운로드를 제거하고 브라우저의 캐싱(Caching)을 활용해 HTTP요청을 줄여나가는 보편적 구현방식을 말한다. 일반적인 파일 형식을 HTML문서, 스크립트, 스타일 등 GZIP과 Deflate에 응용소프트웨어를 사용하여 HTTP응답을 70% 줄일 수 있는 압축기술을 적용하게 된다. 또한 웹의 프로덕션 작성에 있어서도 DOM(: Document Object Model)객체와 CSSOM(: CSS Object Model) 객체의 조작과 변경이 용의해야 유효성 검사에 실패를 낮출 수 있게 된다[12]. 콘텐츠, 스타일링, 사용자 조작 및 유효성 검사는 문서 객체모델(DOM)과 JavaScript를 포함한 스타일 시트(CSSOM)에 의해 순차적으로 수행된다. 순차적 실행을 위해서 JavaScript 엔진을 파일링크로 엮어 DOM과 CSSOM을 종속성 관계 유지하고 제어하기 위한 목적에 의해 사용하게 된다. 순차적 제어방식은 HTML태그에 스크립트 태그가 속해있어 DOM 트리구조를 즉시 중지될 때 이와 동시에 CSSOM를 실행하여 스크립트 태그에 비동기 스크립트를 적용하는 관리체계로 운영된다. 이러한 운영방식을 적용해 웹 페이지의 성능을 높일 수 있는 구조로 편성하게 된다.

3.2.2 결합 적용 기법

인터넷의 통신은 브라우저와 서버사이에 요청과 응답을 주고받는 구조로 이루어진다. 구조의 역할은 클라이언트서버 프로토콜에 의해 GET, POST, HEAD, PUT, DELETE, OPACE, TRACE을 단순하고 일반적인 텍스트형식으로 HTTP요청과 응답의 구조로 적용된다.

HTTP요청과 응답의 경우에 최종 사용자는 응답시간이 10~20%만이 HTML문서를 받게 되고 나머지 응

답시간의 80~90%는 이미지, 스크립트파일, 스타일시트 및 플래시 등을 수행하게 된다. 이러한 응답속도를 줄이기 위해서는 결합 된 스크립트와 스타일 시트기법을 사용하여 전체 응답시간의 최대 50%까지 줄일 수 결합방식을 적용한다[13]. 결합방식은 여러 스크립트 파일을 단일 스크립트로 정리한 후 여러 스타일 시트를 하나의 스타일 시트로 결합하여 배경이미지, 이미지, 폰트, 위치에 관련한 사항들을 CSS속성을 사용하여 필요한 레이아웃의 구성요소에 적용한다. 이러한 결합기법을 통해서 결합되지 않은 스크립트 파일보다 결합기법을 적용한 파일속도가 더 빠르다는 것을 보여 줄 수 있다.

3.2.3 캐시 적용 기법

HTTP캐시는 웹 페이지의 응답시간을 개선시킬 수 있는 또 다른 사용자방식에 속한다. 사용자의 빠른 응답을 위한 방법으로 모든 최신 브라우저를 통해 보편적 채택방식을 적용한다. 일반적으로 서버로드 감소를 줄이기 위해 웹 자원을 사용자 자신에게 복사해 저장하고 필요할 때 서버를 요청하지 않고 저장한 자원을 검색하여 활용할 수 있게 한다. 이런 방법은 웹 페이지의 내용이 자주 변경되는 경우에는 적절하지 않는다. 그래서 ETag(: Entity Tag) 분석을 통해 HEADER요소에 정책을 반영한다.

ETags은 브라우저를 자동으로 인증하고 수정작업이 있을 경우 사용자의 자원에 체크 되어 재배포 되는 구조로 운용된다.

IV. 반응형 웹 페이지 Prototype 설계 구현

이 논문의 주요 목적은 반응형 웹의 인터페이스 구조에 대해 논리를 검토하고 연계하여 프로토타입(Prototype)의 반응형 웹을 구현하는데 있다. 또한 인터페이스 구조에 적합한 개발방식으로 웹 페이지를 디자인하여 융통성 있고 반응성 있게 구현하는 방식에 초점을 두었다.

이 장에서는 설계구현을 개발순서에 의해 “요구사항 분석→전체설계→구현 및 결과”설계 프로세스로 반응형 웹을 설계해 나간다.

4.1 요구사항 분석

요구사항 분석은 웹 페이지의 Prototype 요구사항을 먼저 인터페이스 기반사항에 근거하여 분석하게 된다. “대응적 설계적용 → 최신 HTML5및 CSS5 기술적용 → 디자인(레이아웃) 사용성 원칙적용 → 모든 디바이스지원 구성설계”순차로 분석을 실행한다. 이러한 요구사항의 분석은 본 논문에서 제시한 기술적인 요소들을 근거로 인터페이스에 적합한 사용자 중심의 구조로 적용되어 레이아웃 설계에 반영한다.

4.2 레이아웃 설계

순차적 프로세스 분석이 이루어지면 웹 페이지를 실제 구현하기 위한 사용자 중심의 인터페이스를 구성하는 전체적인 레이아웃을 디자인한다. 레이아웃 설계에 있어 눈에 보이는 것뿐만 아니라 자체 기술을 보유하고 있기 때문에 소프트웨어적 환경에 고려하여 설계에 적용된다. 레이아웃 설계는 해상도를 기준으로 콘텐츠 우선순위에 의해 크게 3가지 모드로 설계가 이루어진다. 데스크 탑 모드, 태블릿 모드, 모바일 모드로 각각의 스타일에 의해 레이아웃을 설계해 나간다.

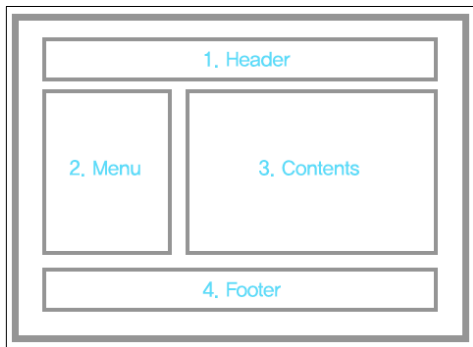


그림 1. 데스크 탑 모드 레이아웃 설계
Fig. 1 A Desktop top mode layout design

데스크 탑 모드는 평균해상도를 1024픽셀에서 1200픽셀까지로 정하고 범위 해상도 안에서 설계된다. 그림 1과 같이 해상도가 1024픽셀 화면보다 높은 해상도를 지닌 데스크 탑 브라우저를 1, 2, 3, 4번으로 표기한다. 표기순은 Header영역, Menu영역, Contents영역, Footer영역으로 구성되도록 설계한다.

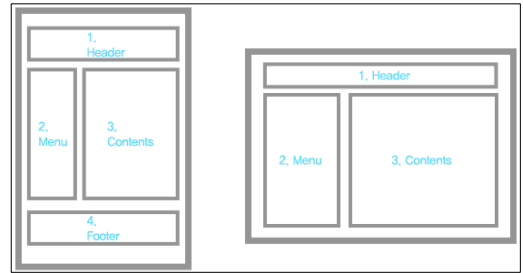


그림 2. 태블릿 모드 레이아웃 설계
Fig. 2 A Design tablet top mode layout

태블릿 모드는 그림 2와 같이 평균해상도를 600픽셀에서 1024픽셀까지 정하고 데스크 탑 모드와 달리 세로와 가로로 구분을 두어 레이아웃을 설계한다. 세로 모드에서는 동적 웹 페이지 영역을 조정하여 1, 2, 3, 4번 전체 영역을 지원할 수 있도록 설계한다. 가로 모드에서는 정적 웹 페이지 영역을 조정하여 1, 2, 3번 부분적 영역을 지원하고 나머지 4번은 스크롤 통해서 영역을 지원한다.

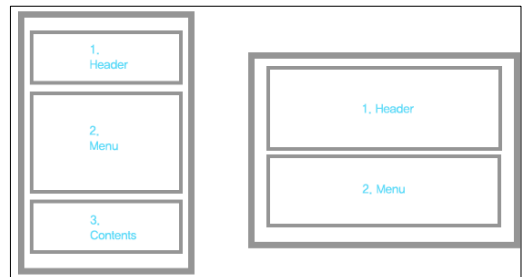


그림 3. 모바일 모드 레이아웃 설계
Fig. 3 A mobile top mode layout

모바일 모드는 그림 3과 같이 세로 모드에 평균 해상도를 320픽셀로 정하고 가로 모드에 평균해상도를 480픽셀로 기준하여 설계한다. 모바일의 화면너비는 데스크 탑 모드의 너비에 비해 평균 70~80% 정도 적기 때문에 데스크 탑 모드나 태블릿 모드처럼 화면 구조상 레이아웃을 구성할 수 없다. 그래서 가로 방향과 세로 방향 모두를 계층 열 방식으로 표현하여 설계한다. 레이아웃 영역을 1, 2, 3, 4번 모두 수직 순서로 정렬하고 여백 모두를 제거하여 전체 레이아웃의 시각적 확보에 중점을 두었다.

4.3 반응 형 구현

인터페이스 구조의 프로세서 기반에 근거하여 레이아웃 설계가 이루어질지라도 무엇보다 “웹 페이지가 어떻게 표현되는지에 대한 기본적인 아이디어에 의해 디자인 되고 완성된다.”라고 이야기하게 된다. 구현에 있어 프로그램 툴로 Dreanweaver를 사용하였다. Drean weaver버전 CS를 사용하여 Dreanweaver에서 자동으로 지원하는 웹 플랫폼을 활용 해 새로운 HTML5프로젝트를 생성하여 구현을 시작했다. 그림 4와 같이 웹 사이트의 기본폴더 구조로 자동생성 되게 된다. 클라이언트는 웹 사이트의 인덱스 페이지가 일반적으로 저장되는 공용_html 폴더의 이름으로 구성되어 CSS, Java Script, HTML5의 각각의 파일을 저장하고 관리된다. 관리되는 전체폴더에 의해 파일 오퍼레이션에 따라 쉽게 파일관리를 할 수 있는 편의성을 갖는다.

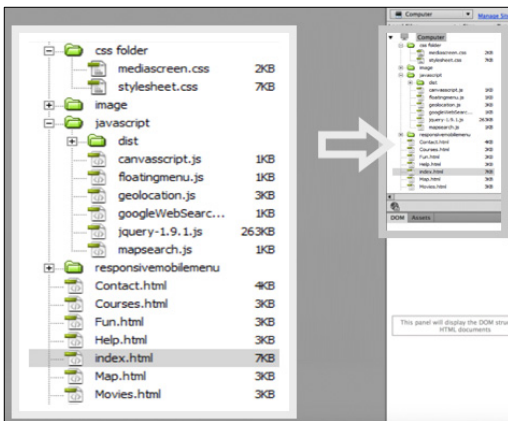


그림 4. 폴더 기본 구조
Fig. 4 A Folder default structure

이러한 툴의 사용은 적응성과 반응성의 레이아웃을 설계할 수 있는 구조로 화면 모드를 제공해 주고 구성요소에 대한 가변적 구문(Fluid Grid, Media Query, Flexible Media)의 코드를 쉬게 작성할 수 있도록 구문 코드를 지원, 자동생성에 도와준다.

콘텐츠 영역의 폭을 백분율 값으로 정의하여 브라우저의 폭이 좁아질 때 영역이 유연해지도록 기본 값을 작성한다. 또한 mediascreen.csv파일 내에 Media Query의 구문을 이용해 화면구성에 따른 레이아웃의 구문을 작성하게 된다. 해상도 기준조건에 따라 미

디어타입과 같이 디바이스의 특성을 이용해 스타일 시트의 적용 범위를 제한하여 기준 폭의 조건 기본식으로 구성된다. 화면의 최소너비가 320픽셀이고 최대 너비가 600픽셀의 화면구성은 모바일 및 소형 태블릿 모드범위로 적용하고 600픽셀부터 1024픽셀은 데스크탑 모드로 지정해 각 디바이스에 대한 유연성을 갖게 했다.

4.4 구현 결과

전체 요건을 이행한 후 웹 페이지를 테스트 해 보았다. 웹 페이지의 일반적 테스트는 최신 브라우저를 공급업체별 브라우저를 각 모드 별 화면크기에 따라 실시하여 그 결과 값을 얻게 되었다.



그림 5. 웹 페이지 화면 구성 데스크 톱 모드
Fig. 5 A Web page screen configuration desktop

데스크 탑 모든 테스트는 그림 5와 같은 화면을 연출했다. 일반적 화면인 데스크 탑 모드는 1024픽셀 보다 클 때 수용 되어진 웹 페이지를 보여준다.

그림 6은 태블릿 모드의 레이아웃 구성을 보여준다. 태블릿 모드는 600픽셀에서 1024픽셀 사이의 디바이스에 대한 웹 페이지를 연출했다.



그림 6. 웹 페이지 화면 구성 태블릿
Fig. 6 A Web page screen configurations table

그림 7은 모바일 모드의 레이아웃 구성을 보여준다. 모바일 모드는 600픽셀 이하일 경우에 디바이스에 대한 웹 화면을 연출했다.



그림 7. 웹 페이지 화면 구성 모바일
Fig. 7 A Web page screen config mobile

이처럼 다양한 콘텐츠 영역을 화면 해상도에 관계 없이 설계를 구현하여 모든 컨테이너가 단일 열에 형식으로 웹을 연출하는 레이아웃의 구성을 적용했다. 태블릿 모드, 모바일 모드 두 모드는 메뉴항목을 화면의 효율성을 감안하여 기본적으로 영역 내에 숨겨 클릭하거나 터치하는 이벤트가 발생할 때 비로소 확장을 갖게 했다.

V. 결 론

이 연구에서는 반응형 웹 페이지를 설계하는 디자인 요소 및 방식, 구현기법, 설계구현에 대한 여러 논의를 통해 구현하게 되었다.

웹 디자인에 설계의 목표는 사용자 경험과 콘텐츠에 영향을 최대한 살려 다양한 기기에 웹 페이지를 제공하는 것에 있다. 웹 콘텐츠들의 자원 등을 확대·축소하지 않고 더 나은 사용자 환경에서 완벽하게 렌더링할 수 있는 구조로 반응형 웹을 구현할 수 있었다. 또한 반응형 웹의 구현은 다양한 버전의 웹 사이트를 만드는 작업량의 부하를 최대한 줄이며 개발비용과 유지하기 위한 추가적인 비용도 감소시킬 수 있는 이점뿐만 아니라 추가적인 한계 값을 피하는 데 도움이 되었다.

반응적인 웹 디자인 원리는 World Wide Web의 새로운 동향 중 하나이며 현재 웹 페이지에 적절한 해결책을 제공할 수 있는 많은 이점과 기회를 가지고 있다. 웹 페이지의 유지관리에 대한 용이성, 저비용 및 작업부하 감소 등 여러 장점이 있지만 반응형 웹 페이지 설계 및 구현하기에 지루한 원칙들이 많다. 또한 반응형 웹은 배너나 팝업창이 필요한 콘텐츠를 제공하는 데 있어 제약이 있어 필요한 목표를 파악하는 것이 우선시해야 한다는 점들을 들 수 있다. 마지막으로 마우스, 키보드, 터치스크린과 같은 제어장치에 해당하는 장치가 없을 경우 적용할 수 없다는 점에서 개발자에게 또 다른 난제가 되고 있다. 그러나 콘텐츠, 디자인, 디바이스, 웹 성능은 사용자 경험 속에서 이어질 수 있기 때문에 더 나은 사용자 경험을 제공할 수 있을 것이 반응형 웹 설계에 있다. 반응성 설계는 급속히 성장하고 있는 분야지만 설계 원칙에 대한 더 나은 해결책의 연구가 필요하다. 그 하나로

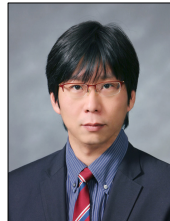
.NET Framework 방식을 적용할 수 있는 방식을 생각해 보게 된다. .NET Framework Web 설계방식은 인터페이스 설계방식과 구조화 시스템 설계방식으로 설계되어 객체에 대한 활용을 높이는 인식성 있는 설계방식을 적용해서 콘텐츠 중심의 분야별 과제로 분석하고 이를 토대로 동적 구조가 아닌 범용적 구조적 기술을 사용할 수 있다는 점에서 매우 흥미를 갖는다.

웹 서비스의 발전은 최신의 개발 솔루션에서 해결책을 찾을 수도 있다. 추세에 따르고 새로운 기술을 사용하면 새로운 기회와 과제가 남게 된다. 또한 개발 기술의 확장에 관심과 표현방식의 연구가 계속 진행되어야 되는 과제로 남게 된다.

References

- [1] K. Developedia, *Korea's 4TH Industrial Revolution*, HansMedia, 2017.
- [2] Cisco. VNI, *Visual Networking Index Global Mobile Data Traffic Forecast for 2013 to 2018*, Cisco®, 2014.
- [3] J. Lee, *Digital culture*, Communication Books, 2013.
- [4] G. Byun, "The Culture Marketing : An Exploratory Approach to Its Concept and Domain," *The j. of Woman & Management*, vol. 3, no. 2, 2011, pp. 71-83.
- [5] B. Martin, *Global Digital Future in Focus 2018*, Cisco®, 2017.
- [6] E. Marcotte, *Responsive Web Design*, alistapart, 2010, URL:<http://alistapart.com/article/responsive-web-design>.
- [7] G. Kim, "Design and Implementation of Adaptive Interaction-based Video Syllabuses," *jkiecs*. vol. 9, no. 9, 2014, pp. 991-998.
- [8] H. Sim, "The Development of Smart TV and Smart Home Platform based on HTML 5," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 12, no. 4, 2017, pp. 663-670.
- [9] J. Seo, C. Kim, "Design and Implementation of Web Contents Management System based on RIA," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 10, no. 10, 2015, pp. 1109-1114.
- [10] S. Hay, *There is no Mobile Web, The haystack*, 2011, URL: <http://www.the-haystack.com/2011/01/07/>.
- [11] S. Krug, *Don't Make Me Think: A Common Sense Approach to Web Usability*, New Riders Publishing, 2014.
- [12] J. Seo, "Implementation of Mobile Social Network System for Shared Contents of Public Service," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 9, no. 9, 2014, pp. 1051-1056.
- [13] S. Sounders, *High Performance Web Sites : Essential Knowledge for Front-End Engineers*, ICT®, 2007.

저자 소개



장승영(Seung-Young Jang)

2009년 남부대학교 대학원 디지털경영정보학과 졸업(공학석사)

2013년 남부대학교 대학원 경영정보학과 공학박사

1996년 ~ 현재 전남과학대학교 정보전산원 근무

※ 관심분야 : Web 개발·기획·분석·정책수립