|  |
| --- |
| **HTML5** |

|  |  |
| --- | --- |
| 적용 분야 | S/W |
| 소속 부서 | 의료영상SW팀 |
| 작성 일자 | 2020. 06. 12. |
| 작성자 | 김재혁 |
| 보안 등급 |  |
| 기여율 | 100% |

목차

[1 개요 3](#_Toc497721509)

[2 연구(검토) 목적 4](#_Toc497721510)

[3 연구(검토) 세부 내용 4](#_Toc497721511)

[4 적용 사례 및 예시 7](#_Toc497721516)

[5 추가 및 보완(개선) 사항 8](#_Toc497721517)

[6 향후 계획 9](#_Toc497721518)

[7 참고 문헌 및 자료 10](#_Toc497721519)

# 개요

|  |  |  |  |
| --- | --- | --- | --- |
| 문서 구분 | ■ 기술 보고 □ 선행 조사 □ 경쟁사 분석 □ 기타 | | |
| 개발 구분 | ■ 개발용 □ 양산용 □ 선행 검토용 ■ 교육용 □ 기타 | | |
| 적용 분야 | ■ 공통 □ 의료기기 □ 산업기기 □ 신사업 □ 기타 | | |
| □ HW ■ SW □ FPGA/FW □ 광학 □ 기구 □ 기타 | | |
| 적용 모델 | QXLink | | |
| 보고서 이름 | Vue.js | | |
| 작성일 | 2020. 03. 29. | 작성 부서 | SW팀 |
| 검토 기간 | 2020. 01. 02. ~ 2020. 03. 8. | | |
| 시험 기간 | 2020. 03. 08. ~ 2020. 03. 29. | | |
| 시험 시료 |  | | |
| 시험 장비  프로그램 | Visual studio 2015 | | |
| 키워드 | Web Development, Font-End | | |
| 요약 정리 | Gdiplus에서 제공하는 Interpolation을 사용하여 왔으나, 성능과 종류의 부족을 느껴 Interpolation에 대해 알아보고 이를 직접 구현해 보기로 한다. | | |

# 연구(검토) 목적

영상의

# 연구(검토) 세부 내용

## HTML5 소개

### 개요

* HTML의 새로운 버전으로 Client Side Thechnology 기술의 중심이 되는 Markup 언어이다.
* Client와 Server와의 통신이 가능하며 이에 대한 부가 기능을 제공함으로써 외부 Active-X와 같은 Plug-in들을 사용하지 않고도 웹서비스를 제공할수 있을 정도로 많은 기능이 추가되었다.
* HTML5 등장으로 CSS3, Javascript API 를 통한 기능의 확장과 결합을 통해 기존 HTML 이상의 의미를 부여하게 되었다.
* Active-X와 같은 Plug-in들을 걷어냄으로서 웹 표준성, 웹 접근성을 높였다.

### 주요 기능

* Device Access: Camera, 동작 센서 등의 H/W 기능을 웹에서 직접 제어
* Connectivity(Web Socket): 웹(Client)에서 Server 측과 직접적인 양방향 통신 가능
* 3D, Graphics & Effects: 다양한 2D 및 3D Graphics 기능 지원
* Styling Effects(CSS3): 글씨체, 색상, 배경 등 다양한 스타일 및 Effect 기능 제공
* Multimedia: Video 및 audio 기능 자체 지원
* Offline & Storage: Network 미지원 환경에서도 Web 이용 가능
* Geo-Location: GPS 없이도 단말기의 지리적인 위치 정보 제공
* Semantics: 웹 자료에 의미를 부여하여 사용자 의도에 맞는 맞춤형 검색 제공
* <body>
* Connectivity(Web Socket): 웹(Client)에서 Server 측과 직접적인 양방향 통신 가능

## CSS

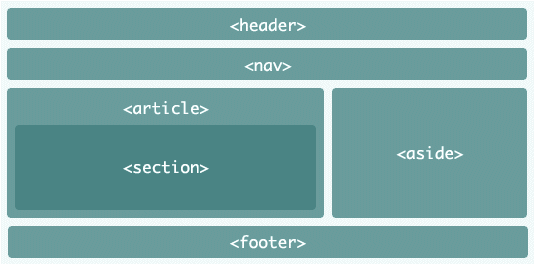
### Display 속성

* + Element를 어떻게 보여줄지를 결정하는 속성
  + 값
    - none: 보이지 않음.
      1. visibility 속성을 hidden으로 설정한 것과 달리, 영역도 차지하지 않는다.
    - block: 블록 박스
      1. <div><p><h><li> Tag들이 해당한다.
      2. 가로영역을 모두 채우며 Block 요소 다음에 등장하는 Tag는 줄바꿈 된다.
    - inline: 인라인 박스
      1. <span><b><i><a> Tag들이 해당한다.
      2. block과 달리 줄바꿈 되지 않고, width와 height를 지정할 수 없다.
      3. Bold, Italic, color, under line등 글자나 문장에 효과를 주기 위해 존재하는 단위
    - inline-block: block과 inline의 중간 형태
      1. 줄 바꿈이 되지 않지만 크기를 지정할 수 있습니다.
  + none Example)

|  |
| --- |
| <style>          .display-none{ display: none }          .invisible{ visibility: hidden }          .block1{ width: 300px; border: 3px solid rgb(95, 27, 27) }          .inline2{width: 200px; /\* 이 값은 무시됩니다 \*/ border: 3px solid #999;}          .inline-block1{display: inline-block; background: #09c; height: 45px;}      </style>  <body>      <div class="display-none">1</div>      <div>2</div>      <div class="invisible">3</div>      <div>4</div>      <div class="block1">5</div>      <p>          Lorem ipsum dolor sit amet, <span>consectetur adipiscing elit</span>,          sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.          Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.          Duis aute irure dolor in <span class="inline2">reprehenderit</span>          in voluptate velit esse cillum dolore eu fugiat nulla pariatur.          Excepteur sint occaecat cupidatat non proident,          sunt in culpa qui officia deserunt mollit anim id est laborum.      </p>      <p>          Lorem ipsum dolor sit amet, <span class="inline-block1">consectetur adipiscing elit</span>,          sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.          Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.          Duis aute irure dolor in <span">reprehenderit</span>          in voluptate velit esse cillum dolore eu fugiat nulla pariatur.          Excepteur sint occaecat cupidatat non proident,          sunt in culpa qui officia deserunt mollit anim id est laborum.      </p>  </body> |

## TAGS

### 구조

* 
* 기본 구조: HTML File을 만들기 위한 기본 생성 Semantic Tag들

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>  </head>  <body>    </body>  </html> |

* <!DOCTYPE html> (DOCTYPE)
  + 기존 DOCTYPE은 SGML(Standard Generalized Markup Language) 기반으로 DTD를 명시해야 했으나, HTML5에서는 Browser가 표준모드로 작동되게 하는 역할만 하면 되어 간소화되었다.
  + DOCTYPE 정의는 Browser를 위해서가 아니라 유효성 검증을 위해 고안된 것으로 DOCTYPE전환을 수행할 때만 주목하게 된다.
  + DOCTYPE이 있다면 표준모드로 작동하게 되고, 없다면 비표준 모드로 작동하게 된다.
* <html lang=”en”>
  + 모든 html 태그들은 <html> </html> 태그 내부에 작성되어야 한다.
  + Language: W3C 규격에 따르면 lang attribute는 Element의Contents와 Text를 포함하고 있는 요소의 Attribute에 대한 기본 언어를 지정한다. 만약 영어로 페이지를 작성하는게 아니라면, 정환한 언어 코드를 지정하는 편이 좋다.
  + 전체 언어 목록은 다음 링크에서 확인
    - <http://www.iana.org/assignments/language-subtag-registry/language-subtag-registry>
* <head>
  + Browser 화면에 직접적으로 표시되지 않으며, 숨은 데이터를 정의하는 Tag들이 들어가게 된다.
  + <body> 태그에서 필요한 내용을 제공할 때 사용된다.

### Head

* <meta>
  + 문서에 대한 정보인 Metadata를 정의할 때 사용한다.
  + Character set, 페이지 설명 요약, 핵심 키워드, 제작자, viewport 등을 정의하는데 사용된다.
  + Attributes
    - charset
      1. HTML 문서의 문자 encoding방식을 정의한다.
      2. Unicode를 위한 UTF-8이 가장 많이 사용된다.
    - content
      1. http-equiv 혹은 name attribute와 관련된 값을 정의한다.
    - http-equiv
      1. Content attribute의 정보/값의 HTTP hedaer를 제공한다.
    - Name
      1. Metadata의 이름을 정의한다.
  + Examples
    - Keword 정의

|  |
| --- |
| <meta name="keywords" content="HTML, CSS, JavaScript"> |

* + - Web Page 설명 정의

|  |
| --- |
| <meta name="description" content="Free Web tutorials for HTML and CSS"> |

* + - Page 제작자 정의

|  |
| --- |
| <meta name="author" content="John Doe"> |

* + - 해당 문서는 30초마다 Refresh 된다.

|  |
| --- |
| <meta http-equiv="refresh" content="30"> |

* + - Website가 모든 Device에서 잘보이기 위한 viewport 설정

|  |
| --- |
| <meta name="viewport" content="width=device-width, initial-scale=1.0"> |

* <title>
  + 웹 페이지의 제목을 나타내는 Tag
  + 본문에 보이지 않으면 Tab 등에서 확인할 수 있다.
  + 사용자에게 문서 제목을 알리는 용도 뿐 아니라, 검색 엔진 등에서 가장 크게 보여지는 Text 이므로 페이지의 특성을 드러내는 제목으로 작성하는 것이 중요하다.
* <style>
  + HTML Document에 간단한 Style Sheet를 적용한다.
  + 어떻게 Browser에서 Render 할지를 정의한다.
  + Example)

|  |
| --- |
| <head> <style>   h1 {color:red;}  p {color:blue;} </style>  </head> |

* <link>
  + 외부 Style Sheet를 Link 한다.
  + 현재 문서와 외부 Resource와의 관계를 정의한다.
  + <link> element는 Empty element로 Attribute만 포함한다.
  + Example)

|  |
| --- |
| <head>   <link rel="stylesheet" href="styles.css"> </head> |

* <script>

### Body (Layout)

* <div>
* <br>
  + 개행 태그
* <span>
  + <div> Tag처럼 특별한 기능을 갖고 있지 않고, CSS와 함께 쓰인다.
  + <div> tag와 차이점은 display 속성이 **block이 아닌 inline이라는 점**.  
    🡪 따라서 <div> 는 줄바꿈이 가능하지만 <span>은 줄바꿈이 되지 않는다.
  + Example)

|  |
| --- |
| <body>  <span style="background-color:red">span1</span>  <span style="background-color:blue">span2</span>  <span style="background-color:green">span3</span>  </body> |

### Body (Component)

* List
  + <ol>
    - 순서가 있는 List를 정의할 때 사용한다.
    - 순서는 숫자 혹은 알파벳으로 리스트화한다.
    - 각 List는 <li> 태그로 정의한다.
    - Example 1)

|  |
| --- |
| <ol>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ol> |

* + - Example 2)

|  |
| --- |
| <ol start="50">   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ol> |

* + <ul>
    - 정렬되지 않은 list를 정의한다. (bulleted: 글머리 기호)
    - Example)

|  |
| --- |
| <ul>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ul> |

* + d
* Button
  + <button>는 클릭할 수 있는 element를 정의한다.
  + text(<i>, <b>, <strong>, <br>, <img>, etc)를 넣을 수 있으나, <input>을 갖는 button을 만든는 것은 불가능하다.
  + 항상 type attribute를 정의해서 browser에게 button의 타입을 말해줘야 한다.
  + Attributes
    - name
    - type
      1. button
      2. submit
      3. reset
    - value
    - autofocus
    - disabled
    - form
    - formaction
    - formenctype
    - formmethod
    - formnovalidate
    - formtarget

### Body (Functions)

* <a>
  + hyperlink 를 정의하는 태그
  + 주요 Attribute 는 href로 link의 destination을 가르킨다.
    - 방문한적 없는 Link는 파란 밑줄이 표시된다.
    - 방문한적 있는 Link는 보라색 밑줄이 표시된다.
    - 활성화된 Link (클릭상태) 는 빨간 밑줄이 표시된다.

### Script 관련

* <template>
  + Page가 Load될 때, 숨겨질 HTML Content들을 가지고 있다.
  + <template> 안에 content는 javascript로 만들어 질 수 있다.
  + 만약 재사용할 HTML code를 가지고 있다면, <templat> tag를 사용할 수 있다.
  + Example)

|  |
| --- |
| <body>      <button onclick="showContent()">Show hidden content</button>      <template>        <h2>Flower</h2>       <img src="smiley.png" width="214" height="204">      </template>        <script>          function showContent() {          var temp = document.getElementsByTagName("template")[0];          var clon = temp.content.cloneNode(true);          document.body.appendChild(clon);          }      </script>  </body> |

# 참고 문헌 및 자료

## HTML 소개

* <https://webdir.tistory.com/85>

## Tags

* <https://www.w3schools.com/tags/tag_meta.asp>

Vieworks Co., Ltd

41-3, Burim-ro, 170beon-gil,

Dongan-gu, Anyang-si, Gyeonggi-do

431-060 Republic of Korea

<http://www.vieworks.com>