
Ajax과 jQuery 기반 약용식물 정보시스템 구현

김해란* · 강소영** · 정희택*** · 한순희***

Implementation of medicinal plant information system using Ajax and jQuery

Hae-ran Kim* · So-young Kang** · Hee-taek Ceong* · Soon-hee Han***

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음
(NIPA-2010-C1090-1021-0012)

요 약

본 논문에서는 비동기 통신에 의해 웹페이지를 다시 읽지 않고 서버로부터 데이터를 가져올 수 있는 Ajax 기술과 경량의 크로스 브라우저 라이브러리인 jQuery를 이용하여 사용자 요구사항을 토대로 약용식물정보시스템을 구현했다. 또한 시스템 구현을 통해 체험한 응답 데이터 이용 방법을 코드로 제시하고 특징을 비교 분석하였다. 구현 시스템은 서버에서 간단한 데이터를 응답 받아 클라이언트 측에서 처리하고 DOM 트리 조작에 의한 웹페이지 변경으로 사용자 상호작용 및 반응속도 향상과 사용자에게 자료 접근 편리성을 제공한다.

ABSTRACT

In this paper, we implemented medicinal plant information system based on user requirement using Ajax technique which can retrieve data from the server asynchronously in the background without refresh webpage and jQuery which is a lightweight cross-browser javascript library. Also, we presented the source code handling the server response data used in the system and compared the features by the type of response data. This system shows the improvement of user interaction and response rates because of a simple response data from the server, client-side data processing and change of web page by manipulating the DOM tree and provides ease of use and convenient data access.

키워드

에이작스, 제이쿼리, 웹 2.0, 약용식물 정보시스템, 웹응용 애플리케이션

Key word

Ajax, jQuery, web 2.0, medicinal plant information system, Web applications

* 전남대학교 디지털컨버전스 (hrankim@jnu.ac.kr)

** 전남대학교 수산생명의학과

*** 전남대학교 디지털컨버전스

접수일자 : 2010. 02. 26

심사완료일자 : 2010. 03. 30

I. 서 론

우리 생활의 중요한 일부가 되어버린 인터넷과 웹은 특히 최근 웹2.0, Ajax 기술 등으로 초창기 웹의 단방향성 단순 정보 제공의 틀을 벗어나 참여, 개방을 통한 공유와 사용자 상호작용이 풍부한 기능 제공을 통해 공유 웹 생태계의 동적인 양방향성 웹으로 진화하고 발전하고 있다.

플랫폼으로서의 웹을 지향하는 웹 2.0의 여러 특징 중 특히 가벼운 프로그래밍 모델의 사례로 Ajax, Open API, RSS 등 기술이 각광받고 있다. 이는 경량의 UI, 경량의 개발 모델, 경량의 비즈니스 모델만이 웹 시장에서 살아남을 수 있음을 시사하며 주목받고 있는 기술이다. 그리고 Ajax 프레임워크로 최근 무서운 상승세로 웹개발 분야에서 이용되는 jQuery가 있다. 이는 개발자들이 브라우저마다 자바 스크립트 코드를 다르게 작성해야 하는 문제, 자바스크립트 기본 함수 기능 부족으로 함수를 묶어서 사용해야 되는 불편 등을 해결해준다. 이는 간결한 방법으로 코드를 구현할 수 있게 하고 개발자들에게 기능에 집중할 수 있도록 한다[1].

따라서 jQuery가 JavaScript와 Ajax 프로그래밍을 단순화할 수 있기에 경량의 프로그래밍 모델에 적합한 기술이라 할 수 있다.

본 논문의 약용식물 정보 시스템은 약용식물정보 데이터베이스를 구축하여 웹상으로 사용자의 상호작용을 통해 빠르고 편리하게 자료에 접근하고자 하는 의도와 경량의 웹 프로그램 기술을 적용하여 보다 쉽고 편리하게 정보에 접근해 가도록 동적이고 반응속도가 빠른 웹 애플리케이션 작성 기술력을 갖고자 하는 의도가 모여져 개발하게 되었다.

본 논문의 구성은 다음과 같다. 1장의 서론에 이어 2장의 관련 연구로 경량의 프로그래밍 모델의 주요 축이 되는 Ajax 과 jQuery 주요 특성과 사용방법 등을 열거한다. 3장에서는 시스템 설계와 구현으로 사용자 요구사항을 토대로 이용된 jQuery와 Ajax을 사용한 구현 내역을 기술한다. 4장에서는 시스템 구현을 통한 서버 응답 데이터 유형 비교 분석으로 유형별 사용 경우 특징을 표로 제시한다. 마지막으로 5장에서 결론과 향후 연구로 끝을 맺는다.

II. 관련 연구

2.1 Ajax

Ajax는 Asynchronous JavaScript and XML의 약어로 용어에 비동기 JavaScript와 XML을 포함하고 있지만 실은 자바스크립트와 XML 이외에도 HTML(또는 XHTML), CSS, DOM 등의 기술과 잘 사용하지 않았던 XMLHttpRequest 기술을 서버와 데이터 교환을 위한 비동기 통신방법으로 활용하는 등 기존의 여러 기술들의 묶음을 지칭한다[2]. 특히, 상호작용이 풍부한 웹 애플리케이션 제작을 위해 기존의 여러 기술들의 조합을 이용하는 웹 개발 기법이라 할 수 있다. Ajax 애플리케이션은 실행을 위한 플랫폼으로 Ajax을 지원하는 모질라 파이어폭스, 인터넷 익스플로러, 오페라, 사파리 등의 다양한 웹브라우저가 이용된다[2]. Ajax은 사용자 요청에 의해 응답된 데이터를 클라이언트 측 브라우저 웹페이지에 렌더링 한 후 DOM트리 조작으로 전체 페이지의 리프레쉬 없이 비동기적으로 데이터 처리가 가능하다. Ajax 기술의 주요 특징은 다음과 같다[1][2][3][4].

- 서버는 간단한 데이터만 리턴하고 처리는 클라이언트에서 하여 데이터 교환 양 감소로 반응속도 향상
- DOM 트리 조작으로 Refresh 없이 웹페이지 안의 내용, 구조, 스타일에 접근 및 변경
- HTML과 CSS 체계적 다름
- 비동기 통신 이용 페이지를 다시 읽지 않고 서버와 데이터 교환
- 교환 데이터 포맷으로 XML, HTML, 텍스트, JSON(JavaScript Serialized Object Notation) 사용
- 마우스 클릭, 마우스 오버, 키 업 등 사용자 이벤트에 의한 비동기 요청

Ajax 기술은 사용자의 요청이 있을 때마다 웹페이지를 다시 읽지 않은 상태에서 즉 페이지 이동 없이 서버와 클라이언트 데이터를 교환할 수 있어 고속으로 화면 전환이 가능하고 소규모 데이터만 요청하여 수신한 처리를 클라이언트에게 위임할 수 있기에 반응 속도 및 성능 향상에 장점이 있다. 그러나 개발 시 스크립트로 작성되기에 디버깅이 용이하지 않고 페이지 이동 없는 통신으

로 보안상의 문제점이 있다[3][4][8].

2.2 jQuery

누구든 사용 가능하도록 공개된 오픈소스 프레임워크인 jQuery는 Ajax 프레임워크로 특히 클라이언트 부분에 집중하는 JavaScript 프레임워크라 할 수 있다. jQuery가 추구하는 목표는 코드를 단순하고 간결하게 유지하고 재사용 가능케 하는 것이다.

특히, 프레임워크를 사용하는 이유는 웹 애플리케이션 개발자들이 브라우저마다 자바 스크립트 코드를 다르게 작성해야 하는 문제, 자바스크립트 기본 함수 기능이 부족하여 자바스크립트 함수를 묶어서 사용해야 하는 불편함의 문제를 덜어주고 기능 개발에 좀 더 집중할 수 있게 하기 때문이다[1].

jQuery는 존재식에 의해 2006년 릴리즈 되어 현재 jQuery 1.4 버전이 릴리즈 되었고 jQuery 공식 웹사이트에서 다운로드 가능하다. 다운 받은 jQuery는 jquery.js 형태의 한 개의 JavaScript 파일로 되어 있다. 이는 HTML 문서를 다루는 자바스크립트 함수를 묶어 놓은 라이브러리로 다양한 브라우저의 호환성을 고려하며 개발에 드는 시간과 노력을 줄여준다. 다음과 같이 웹 페이지에 스크립트를 추가해주면 jQuery를 웹사이트 개발에 사용할 수 있다[5][6][7].

```
<script type='text/javascript' src='./jquery.js'>
</script>
```

무엇보다 jQuery는 CSS 선택자를 이용하여 JavaScript와 DOM 오브젝트의 상호작용을 편리하게 한다. jQuery를 사용하는 기본 방법은 HTML 엘리먼트나 엘리먼트 그룹을 선택해서 jQuery 메서드를 적용하는 것과 \$.가 앞에 붙은 함수를 사용하는 경우이다. 전자는 jQuery 오브젝트를 리턴 하는 반면 후자는 전역 속성이나 전역 함수에 영향을 주는 유틸리티 메서드들로 jQuery 오브젝트 그 자체와 연동되지는 않는다[1].

2.2.1 엘리먼트 선택

jQuery 사용 시 가장 강력한 함수는 \$() 함수이다. 이 함수를 사용하여 웹 문서의 엘리먼트 이름을 즉 CSS 선택자를 이용 선택한다. 또한 기본적인 선택자에 더하여 다양한 선택자 필터들도 제공한다.

표 1은 대표적인 jQuery의 엘리먼트 선택 기능이다. 이외에도 다양한 선택자 필터들이 존재하는데 :contains(text)를 비롯한 내용필터, :hidden, :visible 등의 가시성 필터, 해당 속성을 갖는 엘리먼트를 선택하는 어트리뷰트 필터 등 다양한 필터들이 있어 엘리먼트 선택이 간결하고 강력하다[1][6].

표 1. jQuery 선택자
Table. 1 jQuery selectors

jQuery 표현	의미
\$('#gname')	문서 안의 엘리먼트 ID가 gname 선택
\$(div')	문서 안의 모든 <div> 태그 선택
\$(span.tot')	문서 안의 태그 엘리먼트 중 classname이 .tot 엘리먼트 선택
\$(div:first')	문서 안의 <div> 중 첫 번째 선택
\$(*)	모든 엘리먼트 선택

\$() 함수로 선택한 웹문서의 엘리먼트를 jQuery 객체로 가져와 다음 절의 다양한 메서드를 적용하여 실행시킨다[1].

2.2.2 jQuery 메서드

jQuery의 다양한 메서드는 엘리먼트를 숨기고 드러내기, 엘리먼트에 CSS 속성 적용, 엘리먼트를 추가하고 제거하기, 문서구조의 변화에 따라 특정 엘리먼트 접근하기, 익명의 함수 정의 등을 이용한 특정이벤트 실행하기, Ajax 요청을 실행하기 등의 다양한 메서드가 존재한다[1][5].

- 엘리먼트 숨기고 드러내기 메서드
 - hide(), show() 메서드 등을 이용한다.
 - \$(#gname').hide(); //엘리먼트 숨기기
 - \$(#gname').show(); //엘리먼트 보이기
- CSS 속성 및 태그에 속성 적용 메서드
 - css(), attr(), html(), text() 메서드 이용
- 엘리먼트 추가, 제거하기 메서드
 - before(), after(), remove() 메서드 이용
- 변화하는 문서 구조에 접근 메서드
 - find(), get() 메서드 이용
- 이벤트 메서드
 - click(), keyup() 메서드 이용

- Ajax 요청 post 메서드
 - \$.post() 메서드 이용
- \$.post(url, [data], [callback], [type]);**
- url : 실제로 요청을 처리할 url
 - data : 옵션으로 전송할 값
 - callback : 옵션으로 Ajax 함수의 성공적 수행 후 처리 코드 지정
 - type : 반환자료의 형태인 xml, html, json, text, script를 지정. 생략 시 html 기본

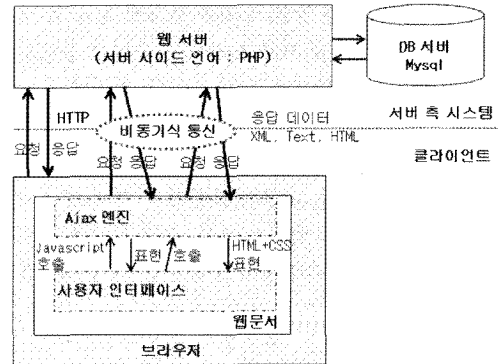


그림 1. 시스템 구조
Fig. 1 System Architecture

III. Ajax과 jQuery기반 약용식물 정보시스템 설계 및 구현

Ajax 애플리케이션의 핵심은 클라이언트 쪽에서 웹 페이지를 다시 읽어 들일 필요 없이 서버 응답으로 간단한 데이터를 받아 클라이언트에서 처리함으로써 반응 속도 및 사용자 상호 작용 향상에 있다. 구현한 시스템에서도 사용자 요구사항을 토대로 3.3과 같은 Ajax의 비동기 통신을 이용하여 중요 기능을 구현 했다.

3.1 사용자 요구사항

약용식물 정보 시스템을 통해 사용자는 보다 쉽고 편리하게 자료에 접근해 갈 수 있도록 다음 사항을 요구 하였다.

- 약용식물 과명에 의한 분류
- 약용식물 작용 값에 의한 분류
- 제시어 자동 완성 등으로 검색 용이
- 약용식물 일반명에 의한 자료 제시
- 권한 있는 사용자만 엑셀자료 다운로드
- 학명, 사용부위, 생약명으로 검색

3.2 시스템 구성 및 사용 기술

구현 시스템에서는 필요한 데이터만을 웹 서버에 요청해서 받은 후, 클라이언트에서 데이터에 대한 처리를 하도록 웹문서를 객체로 표현하는 표준인 DOM, 자바스크립트 등의 Ajax 기술을 적용했다. 또한 비동기식 통신이 가능한 Ajax 기술로 인해 페이지의 전환 없이 사용자가 원하는 정보를 원하는 곳에 보여준다. 그림1의 Ajax 엔진은 서버에 비동기식 통신을 수행하는 JavaScript 코드를 포함한다.

약용식물정보시스템의 서버 구성은 다음과 같다.

- 웹서버 : Aapche
- DB서버 : Mysql
- 서버 쪽 처리 : PHP 언어
- 클라이언트 쪽 처리 : jQuery, JavaScript

3.3 비동기식 통신을 이용한 구현 내역

사용자 요구사항을 통해 검색어 자동완성, 과명, 일반명 분류에 의한 자료 제공, 특정사용자만 엑셀자료 다운로드 가능 등의 항목에 비동기식 통신을 통해 서버로부터 데이터만 응답 받아 클라이언트에서 데이터에 대한 처리를 하였다.

3.3.1 약용식물 과명에 의한 분류 UI

약용식물 과명에 의한 분류는 한 페이지 내에 과명에 대한 색인 구조를 그림 2와 같이 처리 하였다.



그림 2. 약용식물 과명 분류에 의한 UI
Fig. 2 Family category UI of medicinal plant

그림 2에서 사용자 상호 작용은 알파벳 첫 글자 버튼을 클릭하면 해당 첫 글자로 시작되는 과명이 정렬되어 표기되고 페이지 가독성을 위해 10개씩 화면에 표시되게 처리 했다. 특히 비동기 통신을 적용한 부분은 그림 2의 과명을 클릭 시 해당 과명에 속하는 약용식물의 일반명이 표현되는 부분과, 표현된 일반명 중 특정 일반명을 선택하면 작용 값을 표현하는 부분, 그리고 작용값을 선택 시 그림3과 같이 상세 내역이 작용값 위치부터 표현 되도록 한 부분이다.

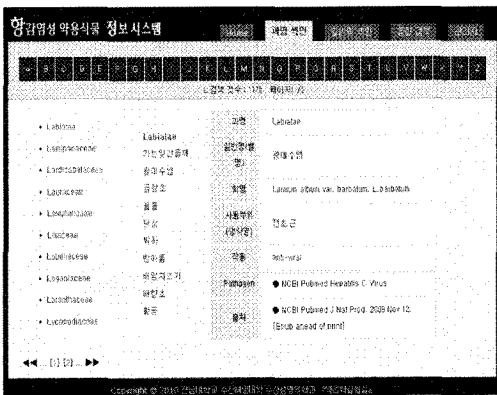


그림 3. 상세내역 UI
Fig. 3 Detail information UI

다음 코드는 과명을 클릭 했을 때 실행하는 jQuery를 포함한 JavaScript 코드이다.

```
function gnameview(cname) {
    $.post('search-gname.php',{keyword : cname},
    function(data) {
        $('#action_div').hide();
        $('#detail_div').hide();
        $('#gname_div').show();
        $('#gname_txt').html('<ul id=aname><\/ul>');
        $('#gname_text > ul').html(data);
    });
}
```

코드에서 html() 메서드가 두 번 사용되는데 첫 번째는 gname_text라는 엘리먼트에 태그 관련 html 내역을 입력하는 것이고 두 번째는 gname_text 바로 아래에

속하는 자식 엘리먼트인 태그에 서버로부터 리턴 받은 html 데이터를 입력한다는 의미이다. 서버사이드 프로그램 search-gname.php로부터 응답 받은 데이터 형태는 html 형태이다.

3.3.2 검색어 자동 완성 UI

제시어 자동 완성은 단어명을 정확히 모르더라도 사용자가 원하는 정보를 찾을 수 있도록 하는 것이다. 여기에선 과명, 작용, 일반명에 대해 입력 글자와 관련된 단어를 그림 4와 같이 자동 제시하여 사용자가 선택하도록 했다.

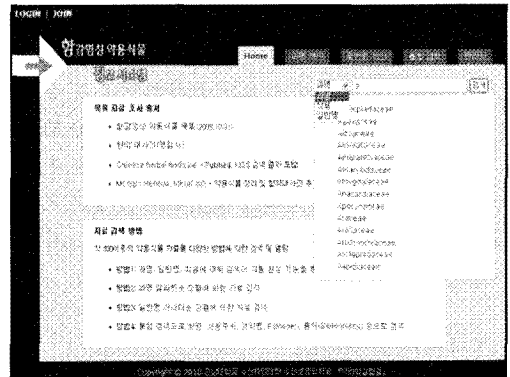


그림 4. 검색어 자동 완성 UI
Fig. 4 UI for search auto-complete

다음은 Ajax에 의해 실행되는 서버사이드 프로그램 코드 일부이다.

```
//한글 처리
$key = iconv("UTF-8","CP949", rawurldecode($key));
:
$result = mysql_query($sque);
while(@$list = mysql_fetch_array($result)) {
    if($keykind == 'gname') {
        //출력 한글 변환
        $gname = iconv( "CP949", "UTF-8", $list[gname] );
        echo "<li >". $gname. "</li>";
    } else {
        echo "<li>". $list[$keykind]. "</li>";
    }
}
:
```

이 부분 구현 시 한글 데이터 처리 문제가 대두 되었는데 자바스크립트에서 Ajax을 통해 넘겨지는 파라미터 데이터 형과 응답 데이터 형은 UTF-8 형태가 기본이다. 따라서 서버 사이드 프로그램에서 요청된 파라미터 값에 대해 한글 처리를 하여 주고 응답 시에도 한글 데이터는 UTF-8로 다시 변환하여 콜백 데이터로 넘겨주어야 한다. 그렇게 처리가 되었을 때 웹문서 데이터 표현에서 글자 깨짐 현상이 없다.

3.3.3 엑셀자료 다운로드 기능 UI

엑셀자료를 다운로드 받기 위해서는 권한 있는 사용자인데서 비밀번호를 아는 경우에만 엑셀자료로 다운로드 가능토록 처리했다. 이 부분에 서버로부터 응답 데이터만 받아 클라이언트 쪽에서 처리했다.

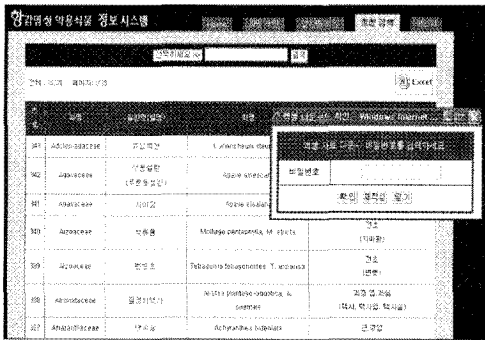


그림 5. 엑셀자료 다운로드 위한 UI
Fig. 5 UI for excel data download

비동기식 통신에 의해 서버로부터 텍스트 형태의 응답을 받아 처리하는 코드이다.

```
function vcheck( ) {
    $.post('pwck.php',{pw:$("#input[name=xpw]").val()},
    function(data) {
        if(data == 'F') {
            window.alert("비밀번호가 틀립니다...");
        } else {
            xhr.submit();
        }
    }, "text" );
}
```

pwck.php 서버 측 프로그램으로부터 권한과 암호를 체크 및 검사 후 "T" 또는 "F" 값의 응답데이터가 이용된다.

3.3.4 약용식물 일반명에 의한 분류 UI

일반명에 대한 분류도 과명 색인의 형태와 유사하게 한 페이지 내에서 색인구조를 갖도록 처리 하였고 한글 자음 버튼을 클릭하면 해당 자음으로 시작되는 이름이 정렬되어 표현 되게 했다. 이곳에서 비동기 통신으로 응답하는 데이터는 XML 형태이다.

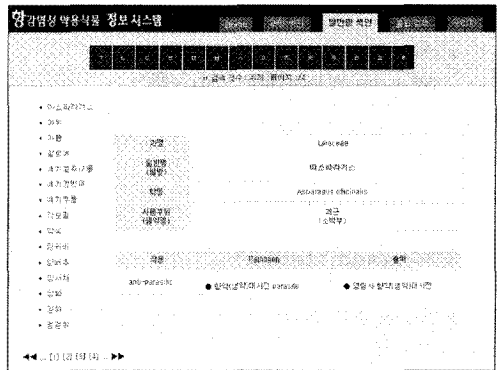


그림 6. 일반명 분류 UI
Fig. 6 UI for common name category

서버 측 data-xml.php로부터 생성되는 XML 데이터의 구조는 다음과 같다.

```
<?xml version="1.0" encoding="utf-8" ?>
<kherbdb>
<knames>
  <kcname>Aizoaceae</kcname>
  <kgname>변행초</kgname>
  :
</knames>
<kagroup>
  <kaction>anti-bacterial</kaction>
  <kpathogen>
    ● 항약(생약)대사전 Saccharomyces spp.
  </kpathogen>
  :
</kagroup>
</kherbdb>
```

다음 코드는 서버로부터 XML 데이터 형태로 응답 받아 웹문서의 DOM 트리 조작으로 브라우저의 갱신 없이 데이터를 표현하고 있다.

```
function detailview(gname,action) {
    $('#action_div').hide();
    $.post('data-xml.php', {key: gname, gu: action},
        function(d){
            var cname = $(d).find("kcname").text();
            var gname = $(d).find("kgname").text();
            :
            var pat = $(d).find("kpathogen").text();
            var chk = patho.indexOf("●",1);
            if(chk!=-1)
                pat = pat.replace(/●/g, "<br/>●");
            :
            // DOM에 값 넣어 주기
            $('#d-cname').text(cname);
            $('#d-gname').text(gname);
            :
            $('#d-pathogen').html(pat);
            :
            $('#detail_div').show();
        }, "xml");
}
```

IV. 시스템 구현을 통한 서버 응답데이터 유형 비교 분석

클라이언트에서 데이터를 서버로 보낼 때 서버와 서버 사이드 언어는 이름, 값 쌍으로 쉽게 인터프리팅 한다. 그러나 서버에서 클라이언트로 데이터를 이름, 값 쌍으로 보내면 클라이언트는 그 응답을 직접 파싱해야 하는 등의 문제점이 있다[6]. 그 대안인 응답 데이터를 XML로 리턴 하는 경우이고, 또한 XML에 비해 데이터양을 줄이고 문자열 객체화 방식을 통해 적은 양의 데이터를 리턴하여 네트워크 대역폭 감소와 응답속도 향상을 꾀할 수 있는 것이 JSON(Javascript Object Notation)이다.

시스템 구현을 통해 알 수 있듯이 검색어 자동완성, 과명, 일반명 분류에 의한 자료 제공, 특정사용자만 엑셀 자료 다운로드 가능 등의 항목에 비동기식 통신을 통해 서버로부터 데이터만 응답 받아 클라이언트에서 데이

터에 대한 처리를 수행함으로써 서버 부하를 줄여 응답 속도 향상을 가져왔고 웹문서를 객체로 표현한 DOM 기술 적용으로 기존 웹 애플리케이션에 비해 사용자 상호 작용이 향상되었다.

또한 시스템 구현 과정을 통해 서버에서 생성하는 응답 데이터 유형별 특징을 표2와 같이 정리할 수 있었다. , <a> 태그 등을 포함하여 생성되는 <html>형태의 응답 데이터는 DOM 핸들링을 통한 사용자 상호작용 시 아주 유용 했고 값이 여러 종류의 데이터를 생성 할 때는 XML 보다는 JSON 형태가 간편하고 코드가 간결했다.

표 2. 서버 응답 데이터 비교 분석
Table. 2 Comparative analysis of server response data

데이터 형태	사용 경우
XML	값이 여러 종류가 사용되는 경우 -단점: 데이터 코드 양이 많음
TEXT	값이 한 종류인 경우
HTML	값이 한 종류이며 html 태그 등과 함께 사용되는 경우 예) 데이터에 , <a> 태그등을 포함하는 경우
JSON	값이 여러 종류이고 XML보다 데이터양을 줄이고자 하는 경우 -특징: 문자열 객체화 방식 이용

응답 데이터의 특징을 표2와 같이 미리 알고 Ajax 웹 애플리케이션의 데이터 모델링 설계에 임한다면 데이터 관리에 매우 효율적일 것이다.

V. 결론 및 향후 연구

최근 웹 개발이 사용자와 상호작용을 통해 페이지 이동 없이 한 페이지 내에서 필요 부분만 갱신하여 네트워크 트래픽을 감소시키는 등의 Ajax 기술을 기반으로 변화와 발전을 거듭해 오고 있지만 아직도 많은 소규모의 웹사이트나 학술 사이트 등에서 웹 표준을 준수하며 Ajax 기술을 적용하고 활용하는 사례들은 극히 일부이다.

따라서 본 논문의 의의를 다음과 같이 요약 한다. 첫째, 사용자 요구사항을 토대로 Ajax 기술을 적용한 약용

식물정보 시스템을 구현한 것이다. 둘째, 사용자 상호 작용 및 응답속도 향상으로 사용자에게 편리성을 제공한 것이다. 셋째, 시스템에서 이용된 서버응답 데이터 활용 내역을 코드로 제시하고 표로 비교 분석 한 것이다. 마지막으로 jQuery를 활용하여 프로그램을 개발함으로써 코드 간결성을 추구한 것이다.

향후 연구는 Ajax기술을 활용한 웹 애플리케이션 개발 시 설계 단계부터 풍부한 사용자 인터페이스를 제공하도록 RIA기술 등을 적용한 기본 설계 방법론을 정립할 계획이다.

참고문헌

- [1] 배민호, '쉽고 빠른 웹개발 Django', 인사이트, 2009년
- [2] <http://ko.wikipedia.org/wiki/Ajax>
- [3] 최지호, 'Ajax 및 PHP를 기반으로 한 SSL의 구현', 연세대학교 석사학위논문, 2008년
- [4] 전국현, 문석재, 엄영현, 국윤규, 정계동, 최영근, 'AJAX를 이용한 실시간 예약 시스템 설계', 한국정보과학회논문집, 33권 2호, 2006
- [5] <http://en.wikipedia.org/wiki/JQuery>
- [6] <http://www.ibm.com/developerworks/kr/series/>
- [7] <http://jquery.com/>
- [8] 이주래, 'Ajax 기반의 위치 추적 서비스 설계 및 구현', 고려대학교 컴퓨터정보통신대학원 석사학위논문, 2008년
- [9] <http://earc.chonnam.ac.kr/herb>

저자소개



김해란(Hae-Ran Kim)

1993.2 전남대학교 전산통계학과
학사
1996.2 전남대학교 전산통계학과
석사

2008년 3월 ~ 현재 전남대학교 디지털컨버전스
박사과정

※관심분야: 시맨틱 웹, 디지털 영상처리, RFID



강소영(So-Young Kang)

1996.2 서울대학교 약학과 학사
1998.2 서울대학교 약학과 석사
2003.8 서울대학교 약학과 박사
2004.3 ~ 현재 전남대학교
수산생명의학과 부교수

※관심분야: 생리활성물질, 수산용의약품



정희택(Hee-Taek Ceong)

1992.2 전남대학교 전산통계학과
학사
1995.2 전남대학교 전산통계학과
석사

1999.8 전남대학교 전산통계학과 박사

1999년 9월~현재 전남대학교 디지털컨버전스 부교수

※관심분야: 멀티미디어, 데이터마이닝, IT융합



한순희(Soon-Hee Han)

1983.2 경북대학교 전자공학과
학사
1985.2 광운대학교 전자계산학
석사

1993.2 광운대학교 전자계산학 박사

1992년 ~ 현재 전남대학교 디지털컨버전스 교수

※관심분야: 모바일 웹, 이동통신, 컴파일러, RFID