

Ajax MVC 기반 일정관리프로그램

Schedule Management Program based on Ajax MVC

박진철, 이영경, 김태성, 장해용, 신동석
동명대학교 컴퓨터공학과

Park Jin-Chul, Lee Yeong-Kyeong, Kim Tea-Sung,
Jang Hea-Yong, Shin Dong-Suk
Dept. of Computer Engineering, TongMyong Univ.

요약

최근 Web2.0을 이용하여 다양한 웹 어플리케이션이 개발 및 서비스 되고 있다. 본 논문에서는 기존의 서버스크립트 언어로 구현된 일정관리프로그램의 단점을 보완하기 위하여 MVC 패턴을 적용한 Ajax 기반의 개인 및 그룹용 일정관리프로그램을 Java Servlet으로 구현하였다. 또한, OPEN API를 활용한 Mash-up을 통해 사용자들에게 다양한 부가 서비스를 제공함으로써 어플리케이션의 활용도를 제고하였다.

Abstract

Recently, variety Web applications which is used Web2.0 are being developed and serviced. In this paper, we implemented Schedule Management Program for the individual and group based on Ajax applications, which apply the MVC design pattern with Java Servlet in order to complement Schedule Management Program which is developed with server-script languages. Also, we improved utilization of applications to provide various additional service through mash-ups built by combing APIs.

I. 서론

인터넷 기술의 급격한 발전과 더불어 웹 브라우저를 통한 다양한 서비스들이 속속 등장하고 있으며, 새로운 개념의 인터페이스를 적용한 비즈니스 모델 및 제품개발이 이루어지고 있다[1]. 그러나 아직까지 대부분의 웹 브라우저를 통한 인터페이스는 사용자의 Activity에 의해 데이터 Request가 발생되고, 웹 서버는 사용자의 Request에 대한 Response를 반복하며 단위 워크플로우를 처리한다[2]. 특히 복잡한 대화형 웹 어플리케이션 개발 및 실시간 정보교환 시스템 구축을 위해 ActiveX, Flash, Java Applet 등을 이용하고 있어 시스템이 정상적으로 작동하기 위해서는 필수적으로 사용자가 Plug-in을 설치해야 하는 불편함이 존재한다.

본 논문에서는 Ajax MVC 기반 일정관리프로그램 구현을 통해 Plug-in 설치 없이 모든 브라우저에서 동일하게 작동하고, 확장성과 데이터 교환이 용이하며 빠른 응답이 보장되는 플랫폼을 구축하였다. 여기서는 Ajax의 XMLHttpRequest 객체를 이용하여 페이지 Refresh 없이 실시간 데이터의 변동을 보여주고, MVC 패턴을 적용, 웹 어플리케이션에 대한 확장성과 유지보수가 용이한 효율적인 개발환경을 제시하였다.

1. Ajax 와 MVC 패턴

1.1 Ajax

Ajax(Asynchronous JavaScript and XML)는 서버로의 비동기 통신 기술과 동적 클라이언트 스크립팅 기법을 하나로 묶은 것이라 할 수 있다.

- XMLHttpRequest 객체가 Web Server와 직접 통신함
- Web Server의 응답 결과는 HTML문서가 아닌 필요한 데이터를 담고 있는 XML이나 텍스트 문서임
- 페이지를 Reload 하지 않고 필요한 모듈만 갱신함[2]

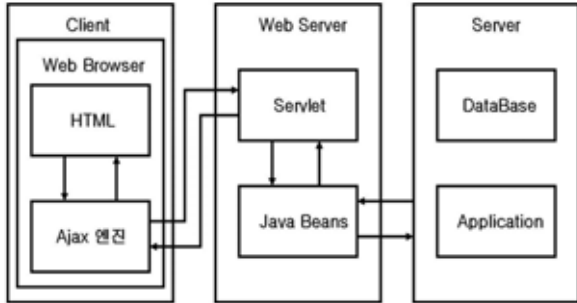
1.2 MVC 패턴

MVC(Model-View-Controller)는 사용자에게 프리젠테이션을 제공하는 뷰, 비즈니스 로직을 처리하는 모델 과 모델을 제어하는 컨트롤로 분리하여 어플리케이션을 개발하는 형태이다. 이는 MVC 기능을 하나의 어플리케이션에 작성함에 따른 소스의 복잡성과 유지보수의 어려움을 피하기 위한 것이며, 비즈니스 로직을 모델 중심으로 접근하여 효율성을 최대화 하려는 의도를 담고 있다[3].

2. Ajax MVC 기반의 Application

Ajax MVC 패턴은 브라우저의 요청을 Servlet이 받아서 데이터 처리 로직을 호출, 처리하여 브라우저로 응답한다. 이런 일련의 과정을 비동기 통신으로 데이터만 처리하므로 실행속

도도 빠르고 사용자가 서버에서 처리가 완료될 때까지 기다리지 않아도 된다[4].



▶▶ 그림 1. Ajax MVC 패턴 구조

앞서 언급된 Ajax MVC 패턴을 기반으로 다양한 어플리케이션들이 개발되고 있다.

[표 1] 대표적인 일정관리프로그램[5][6][7]

제품	제공회사	특징
Google Calendar	Google	다중 일정 생성, 이메일과의 연동.
CalendarHub	CalendarHub	개인 일정 관리
Zimbra	Zimbra	개인 일정 관리, RSS, 스카이프, 시멘틱 웹 기술.

현재 사용되고 있는 이러한 제품들은 개인의 일정관리를 기본적으로 제공하고 스카이프나 시멘틱 웹 등 다양한 기술들을 제공한다. 본 논문에서는 개인일정관리 기능 이외에 그룹을 생성, 관리 가능하며, 개인이 그룹의 일정을 포스팅이 가능하게 만들어 일정관리에 편리함을 더하였다.

또한 네이버와 Google API(Application Programming Interface)를 Mash-up하여 맵 정보와 날씨를 개인과 그룹 일정관리에서 확인 가능하게 하였다.

II. Ajax MVC 기반 일정관리프로그램 개발

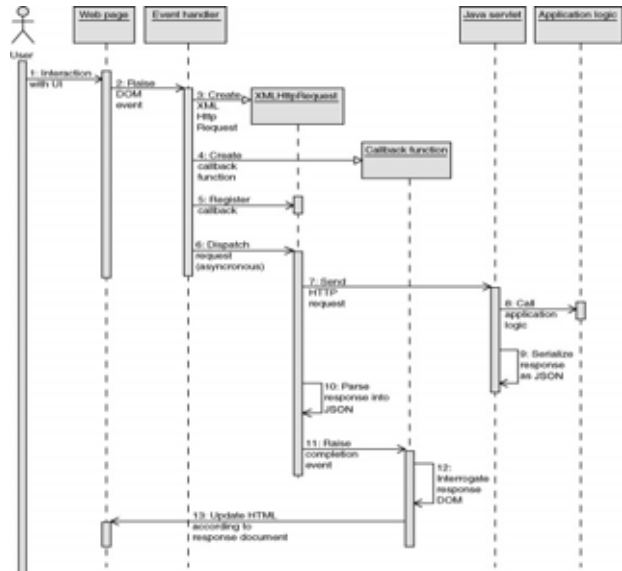
본 논문에서 개발한 일정관리프로그램은 Ajax MVC 기반으로 구현하였으며, 네이버와 Google의 OPEN API를 Mash-up 하여 사용자의 편의성을 높였다[8].

1. 구조

전체적인 구조는 사용자들이 일정을 입력 및 수정, 삭제할 수 있는 User Interface 부분과 Server와 Ajax 통신을 위한 Ajax Engine, DataBase를 가진다. Ajax Engine은 클라

이언트에서 요구를 서버측의 Data Base에 데이터 저장 및 읽기를 처리하여 그 결과를 Ajax Engine을 통해 페이지 변환 없이 User Interface에서ダイナ믹하게 동작할 수 있게 하였다. Ajax Engine의 데이터 통신은 JSON(JavaScript Object Notation)을 사용하여 데이터에 자료형이 부여되어 수동적인 해석과정과 파싱 과정이 없다는 장점을 가진다[9].

Servlet을 사용하여 Server Script Language의 사용을 줄이고 MVC 패턴을 통한 View, Model, Controller의 분리는 페이지 각각의 모듈들의 로딩 시간을 줄이고 보안성 및 확장성이 좋아 클라이언트와 개발자 모두 편의성을 높일 수 있었다. 일정관리 시스템의 데이터 시퀀스 다이어그램은 [그림 2]와 같다.



▶▶ 그림 2. 일정관리 데이터 시퀀스 다이어그램

[표 2] Ajax 통신 함수의 구문

```
function modifyDone(){
    var passwd = document.getElementById("passwd").value;
    createRequest();
    var url = "ModifyMember.do";
    request.open("POST",url,true);
    request.onreadystatechange = updateModify;
    request.setRequestHeader("Content-Type","application/x-www-form-urlencoded");
    request.send("passwd="+escape(passwd));
}

function updateModify(){
    if(request.readyState==4){
        if(request.status==200){
            var jsonData = eval('(' + request.responseText + ')');
            var result = jsonData.result;
            var gid = jsonData.id;
            var gname = jsonData.name;
            var gpasswd = jsonData.passwd;
            if(result == true){
                document.getElementById("oid").innerText=gid;
                document.getElementById("opasswd").innerText=gpasswd;
                document.getElementById("oname").innerText=gname;
                modifyInfoView();
            }
        }
    }
}
```

일정관리프로그램에서 JSON객체를 받아서 처리하는 Ajax 통신 함수 구문을 기술하면 [표 2]와 같다. XML HttpRequest

객체를 생성하여 CallBack함수를 호출하고 DOM 트리를 조작하여 페이지 Refresh 없이 해당 데이터를 사용자에게 보여 줄 수 있다

2. 사용자 기능 정의

2.1 개인 및 그룹 일정관리프로그램 기능 정의

본 프로그램은 타 일정관리프로그램들의 기능에 그룹관리 기능을 추가하여 그룹의 일정 관리에 유용하게 하였다.

- 개인 및 그룹 일정 입력/수정/삭제 기능
- 그룹 생성 기능
- 그룹 일정을 개인의 일정에 적용 시키는 포스팅 기능

2.2 OPEN API Mash-up 기능 정의

- 네이버 지도 API를 사용해서 일정이 행해지는 장소를 표시해주는 기능[10]
- Google 날씨 API를 사용해서 금일부터 4일간의 날씨를 표시해주는 기능[11]

3. 개발 환경

일정관리프로그램 개발을 위한 개발 환경은 아래와 같다.

3.1 Client

- Web Browser : Internet Explorer 6.0, FireFox 1.5
- Library : Prototype JavaScript Library
Scriptaculous JavaScript Library
- Language : JavaScript, Css, HTML

3.2 Server

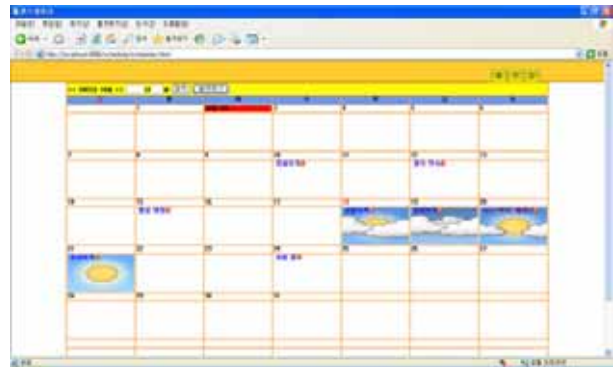
- Web Server : Apathe 2.0, Tomcat 5.5
- Language : Java 5.0 Servlet
- DataBase : MySQL 5.0

4. 개발 결과

User Interface는 [그림 3]과 같이 달력의 형태를 가진다. [그림 4],[그림 5] 처럼 달력상의 기본적인 클릭만으로도 일정의 입력 및 수정, 확인이 편리하도록 설계하였다.

4.1 기본 화면

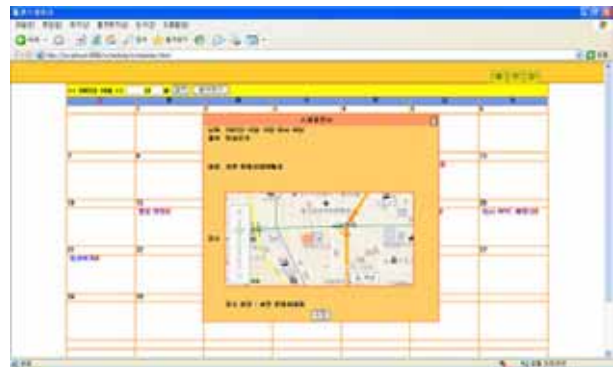
일정관리프로그램의 로그인 후 스케줄러를 생성한 화면이다. 일정이 입력되면 해당 일자에 간단하게 제목이 표시된다. 상위 탭에서 날씨 보기를 클릭 하면 Google의 날씨 API를 Mash-up 하여 4일 간의 날씨를 달력 위에 표시한다.



▶▶ 그림 3. 기본 화면

4.2 입력된 일정 Display

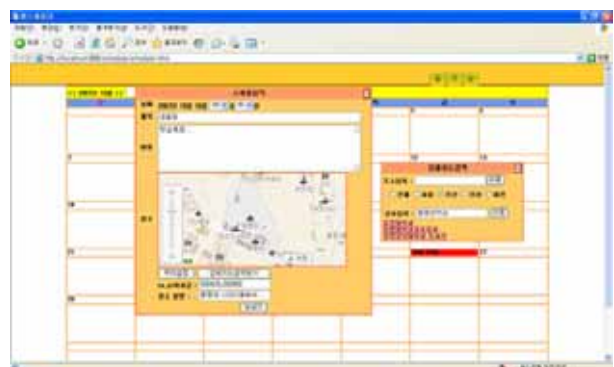
미리 입력된 일정을 확인할 때, 달력 상에서 해당 일자에 입력되어진 일정을 클릭하게 되면 새로운 레이어가 하나 생겨나면서 일정 정보를 Display 한다. Display이 된 정보는 날짜와 제목, 내용, 그리고 일정과 관련된 위치를 표시한다. 여기서 위치 표현은 네이버 지도 API를 Mash-up하여 활용하였다.



▶▶ 그림 4. 일정 정보 Display 화면

4.3 일정 입력 Form

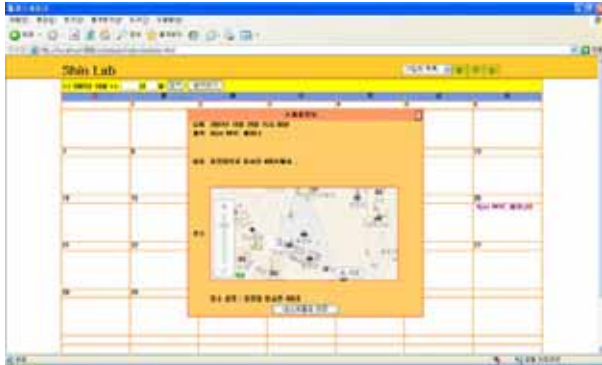
일정을 입력할 때의 Form은 기본적으로 날짜, 제목, 내용 그리고 위치를 입력할 수 있다. 네이버 지도 API의 사용의 편리성을 더하기 위해 상세 위치 검색 기능을 추가하여 원하는 위치를 더욱 더 빠르고 편리하게 검색할 수 있다.



▶▶ 그림 5. 일정 입력 Form 화면

4.4 클럽 일정 포스팅

일정관리프로그램에 가입한 사람 누구라도 불편없이 자신의 필요에 의해서 그룹을 생성 할 수 있으며, 그룹에 가입된 다른 사용자들이 그룹의 일정을 확인하고 참여 하고 싶은 것이 있다면 클릭 한번으로 자신의 개인 일정관리로 포스팅이 가능하다.



▶▶ 그림 6. 그룹 일정 포스팅 화면

[7] <http://www.zimbra.com/>

[8] [http://en.wikipedia.org/wiki/Mashup_\(web_application_hybrid\)](http://en.wikipedia.org/wiki/Mashup_(web_application_hybrid))

[9] <http://www.json.org/>

[10] <http://cafe.naver.com/openapi.cafe>

[11] <http://code.google.com/apis/calendar/overview.html>

III. 결 론

본 논문에서는 Ajax MVC 패턴과 OPEN API를 Mash-up 하여 일정관리프로그램을 개발하였으며, 이를 통해 기존의 웹 서비스로 제공하던 것과는 다른 편리하고 확장성이 뛰어난 웹 어플리케이션을 제공하도록 하였다. 또한 기존의 일정관리프로그램들이 제공하던 개인 위주의 범주를 넘어 다수의 사용자들이 활용할 수 있는 그룹 일정관리 기능을 추가하였다.

현재는 지도상의 위치만을 Display 하는 기능만을 가지고 있지만 추후에는 일정이 행해지는 목적지에 도착하기까지의 길 안내 및 교통편을 상세히 알려줄 수 있도록 기능을 확장할 것이며 그룹의 관리자가 자신의 그룹에 속한 인원들의 일정을 간략하게 확인하여 그들의 활용 가능한 시간들을 추출해 낼 수 있는 기능과 현재 널리 사용되어 지고 있는 카페 형식을 활용할 수 있는 프로그램으로 기능을 확장해 나갈 것이다.

■ 참고 문헌 ■

- [1] 홍성원, "X인터넷과 웹2.0", 한국 데이터베이스 진흥센터 월간 잡지, 디지털콘텐츠 9월호, pp.50-51, 2006.
- [2] <http://www.dbguide.net/blog/blog.jsp?urlid=sophia74>
- [3] 김영보, "Ajax 활용", pp.21-28, ORACLE KOREA Magazine 2006 Summer, 2006.
- [4] 김영보, "프로토타입 완전분석 Ajax prototype.js", pp.24-25, 위키 북스, 2007.
- [5] <http://www.google.com/intl/ko/googlecalendar/tour.html>
- [6] <http://www.calendarhub.com/>