

MAPINUS

HomePage

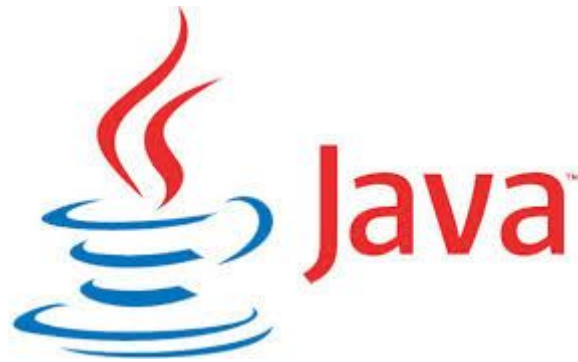
작성자 : 이 현 빈

Content

1. 개발 환경.....	2
1-1 개발 언어.....	2
1-2 서버.....	2
1-3 프레임워크.....	3
2. 홈페이지 구조.....	4
3. 사용 서비스	5
3-1 Spring.....	5
3-1-1 Mail 서비스.....	5
3-1-2 다국어 지원.....	6
3-1-3 팝업창 서비스.....	7
3-2 Bootstrap.....	9
3-3 반응형 서비스.....	13
3-4 기타.....	16
3-4-1 Loader.....	16
3-4-2 Favicon.....	16

1. 개발환경

1-1 개발 언어



자바는 썬 마이크로시스템즈의 제임스 고슬링과 다른 연구원들이 개발한 객체 지향적 프로그래밍 언어입니다. 1991년 그린 프로젝트라는 이름으로 시작해 1995년에 발표했습니다. 처음에는 가전제품 내에 탑재해 동작하는 프로그램을 위해 개발했지만 현재 웹 애플리케이션 개발에 가장 많이 사용하는 언어 가운데 하나이고, 모바일 기기용 소프트웨어 개발에도 널리 사용하고 있다. 현재 버전 10까지 출시했습니다.

1-2 서버



Apache Tomcat

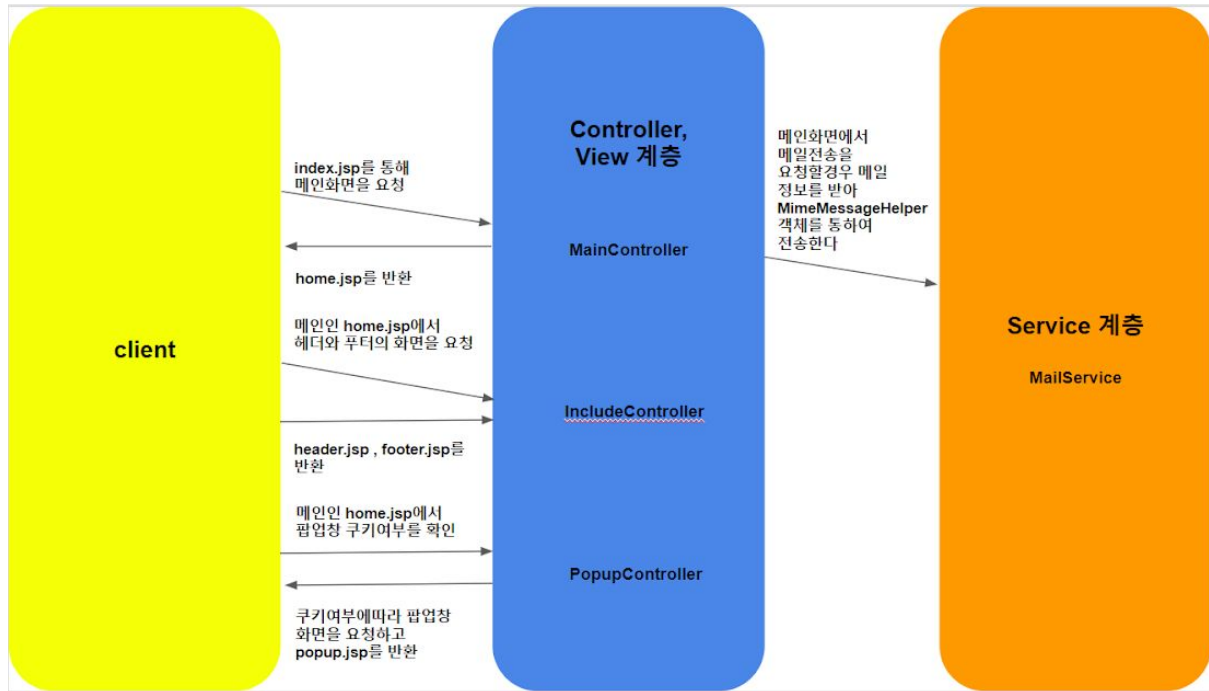
아파치 톰캣은 아파치 소프트웨어 재단에서 개발한 서블릿 컨테이너만 있는 웹 애플리케이션 서버입니다. 톰캣은 웹 서버와 연동하여 실행할 수 있는 자바 환경을 제공하여 자바 서버 페이지와 자바 서블릿이 실행할 수 있는 환경을 제공하고 있습니다. 톰캣은 관리툴을 통해 설정을 변경할 수 있지만, XML 파일을 편집하여 설정할 수도 있습니다. 그리고, 톰캣은 HTTP 서버도 자체 내장하기도 합니다.

1-3 프레임워크



스프링 프레임워크는 자바 플랫폼을 위한 오픈소스 애플리케이션 프레임워크로서 간단히 스프링이라고도 불립니다. 동적인 웹 사이트를 개발하기 위한 여러 가지 서비스를 제공하고 있습니다. 대한민국 공공기관의 웹 서비스 개발 시 사용을 권장하고 있는 전자정부 표준프레임워크의 기반 기술로서 쓰이고 있습니다.

2. 홈페이지 구조

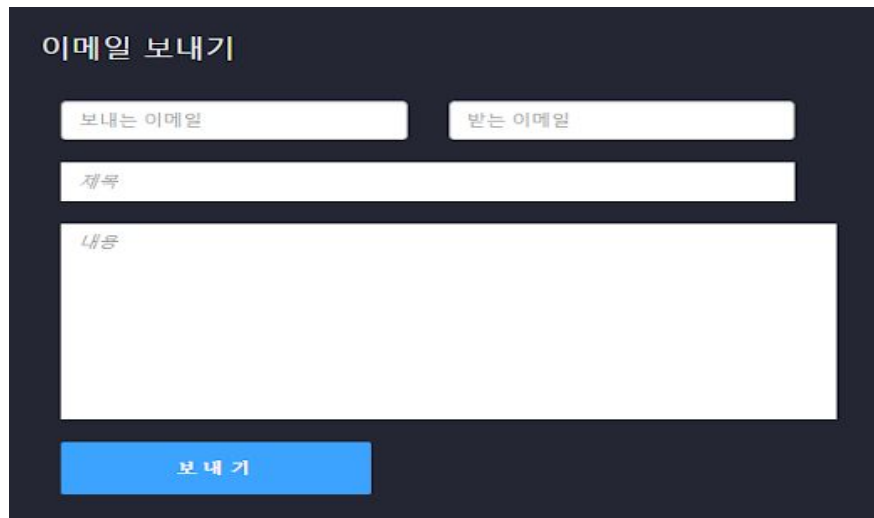


본 홈페이지는 스프링 프레임워크 형태의 싱글 홈페이지 입니다. `index.jsp`를 시작으로 메인화면에서 요청이 들어오고 컨트롤러를 거쳐 메인화면인 `home.jsp`를 반환합니다. 헤더와 푸터를 확장성을 위하여 메인화면과 구분하였습니다. 헤더와 푸터는 다른 컨트롤러를 통하여 `header.jsp`와 `footer.jsp`를 반환합니다. 메인화면에서 메일 요청이 들어왔을시 메일 서비스를 통해 메일을 보내고 새로운 메인화면을 반환합니다. 메인화면을 열었을시 쿠키체크를 통해 `popup`창의 실행여부를 파악하고 쿠키가 존재할시에 팝업창을 요청하여 실행합니다.

3. 사용 서비스

3-1 Spring

3-1-1 mail 서비스



이메일 보내기

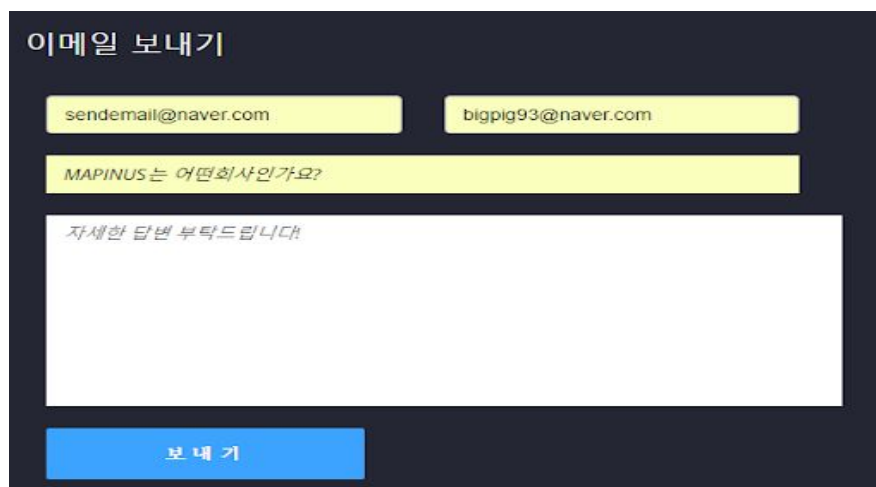
보내는 이메일 받는 이메일

제목

내용

보 내 기

그림 1 메일을 보내기 전 입력창



이메일 보내기


sendemail@naver.com bigpig93@naver.com

MAPINUS는 어떤회사인가요?

자세한 답변 부탁드립니다

보 내 기

그림 2 메일을 보내기 위한 데이터 입력

☆ MAPINUS는 어떤회사인가요? 

보낸 사람 ☆ <lhbdong@gmail.com>

받는 사람 <bigpig93@naver.com>

질문 이메일 : sendemail@naver.com

내용 : 자세한 답변 부탁드립니다!

그림 3 메일을 보낸 결과

그림 1은 메일을 보내기 위한 입력폼으로 입력여부를 체크하여 그림2와 같이 메일을 올바르게 입력했을 경우에 메일이 보내집니다. 입력한 데이터를 바탕으로 메일정보를 MailService의 JavaMailSender 객체를 통하여 전송합니다. mail-context.xml 에서 JavaMailSender에 대한 설정을 변경하여 보낸사람을 바꿀수 있습니다.

3-1-2 다국어 지원

그림 4 언어를 선택하는 Select 박스



그림 5 한국어버전

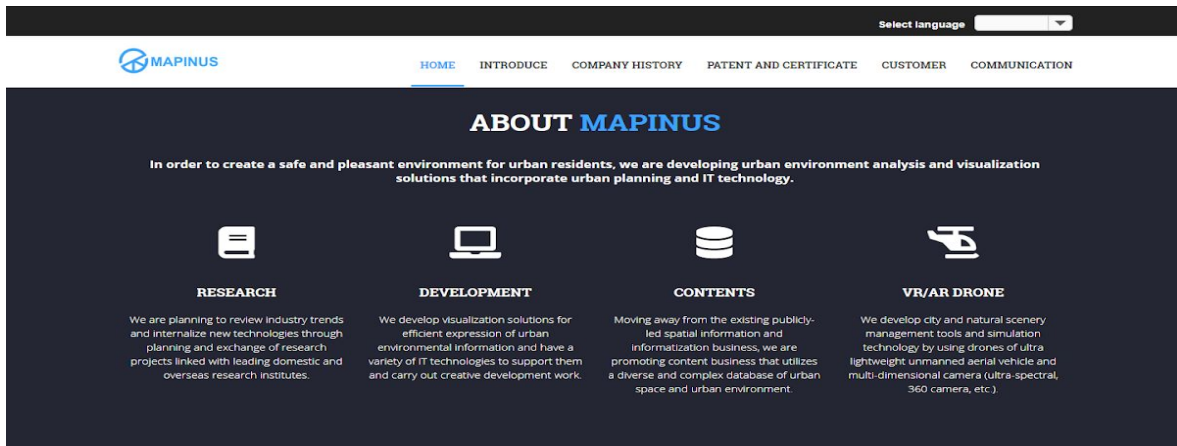


그림 6 영어 버전



그림 7 중국어 버전



그림 8 일본어 버전

action-servlet.xml을 살펴보면 message에 대한 설정과 locale을 변경하기 위한 파라미터를 변경하는 코드가 입력되어 있습니다. 여기서 message란 각 언어별로 텍스트를 분리한 파일들을 의미하고 locale은 각 message를 선택하기 위한 속성명이라 생각하면 됩니다. 예를들어 파라미터로 ?lang=ko를 통해 ko라는 값이 들어오면 message-ko.properties 라는

파일을 찾게 되고 해당파일의 언어인 한국어 텍스트를 읽어 jsp파일에 텍스트를 뿌려줍니다. 그림 5 ~ 그림 8 은 각 언어가 설정된 일부 화면을 캡처한 것입니다.

3-1-3 팝업창 서비스



그림 9 팝업창 예시

간단하게 팝업창 예시를 만든 것으로 활용하기 위해서는 팝업창의 그림의 path를 바꾸면됩니다. 하단의 체크박스를 체크하고 닫을시 메인화면을 다시 호출하면 팝업창을 그 날 하루동안은 열리지 않습니다.

3-2 Bootstrap

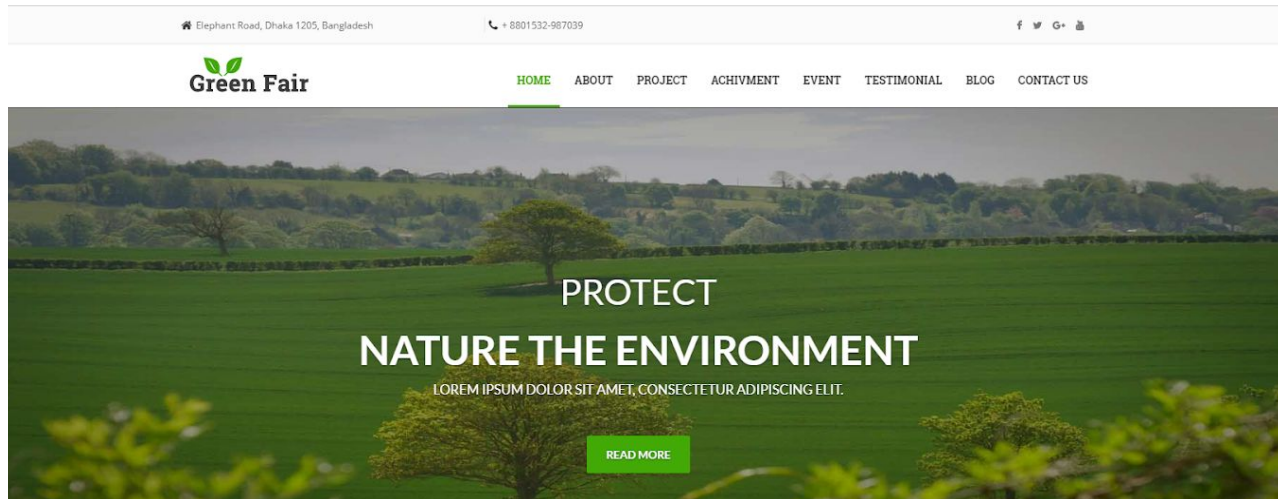


그림 9 메인 Template

그림 9는 메인으로 사용된 무료 Template입니다. 헤더와 푸터와 일부 Bootstrap을 사용하였습니다.

참고 URL : <https://www.free-css.com/free-css-templates/page229/greenfair>

그림 10 Selectbox

언어를 설정하는 선택박스는 wSelect를 사용하였습니다. 이 선택 박스는 기존의 선택박스와는 달리 그림을 추가로 넣을 수 있습니다. 국기를 표시하기 위해 이미지 선택 박스를 사용하였습니다.

참고 URL : <https://github.com/websanova/wSelect>



그림 11 Main Slider

메인 슬라이더는 다른 슬라이더와 차이를 주기 위해 Bootsnipp의 Preview Carousel 을 이용 하였습니다. 중간 밑에 보시면 슬라이더의 전체를 썸네일로 표현 하였습니다.

참고 URL :

<https://bootsnipp.com/snippets/featured/responsive-thumb-preview-in-carousel-indicators>



그림 12 기술 소개

기술 소개에 쓰인 Bootstrap은 Main Template에 쓰인 Bootstrap 으로 아이콘은 Fontawesome을 사용하였으며 아이콘으로 마우스를 가져갔을시에 Hover 이벤트가 발생하게 됩니다.



그림 13 회사연혁

회사 연혁을 테이블 형식의 단순한 UI보다는 Bootsnipp의 Timeline을 이용하여 작성했습니다.

참고 URL : <https://bootsnipp.com/snippets/featured/timeline-dotted>



그림 14 특허 및 인증서

특허 및 인증서는 계속적인 추가가 될 가능성이 많습니다. 추가시 페이지가 무한히 길어질수 있는 것을 방지하기 위하여 슬라이더를 넣었습니다. 마우스를 가져다가 대면 Hover 이벤트를 추가하였습니다. Hover 이벤트 또한 Fancybox를 이용하여 사용 하였습니다. 기본적 슬라이드는 Bootsnipp의 슬라이드를 이용 하였습니다.

참고 URL : <https://github.com/fancyapps/fancybox>
-그림 관련된 이벤트와 css에 사용

<https://bootsnipp.com/snippets/featured/media-slider-carousel-bs3>

-슬라이더에 관련된 이벤트와 css에 사용

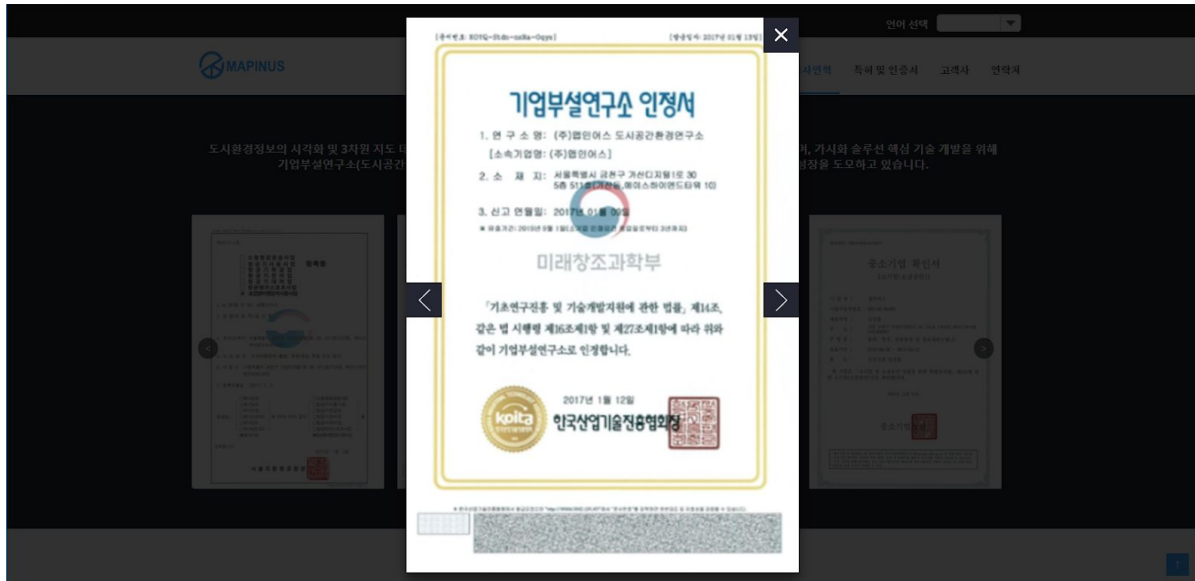
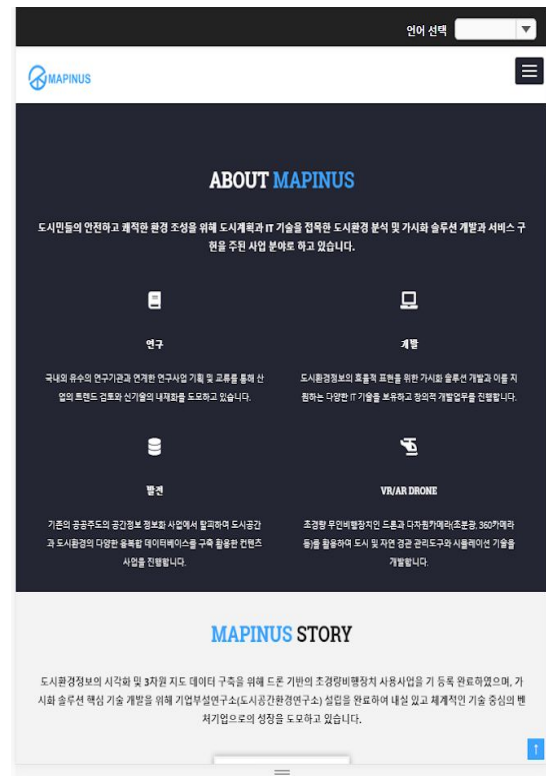


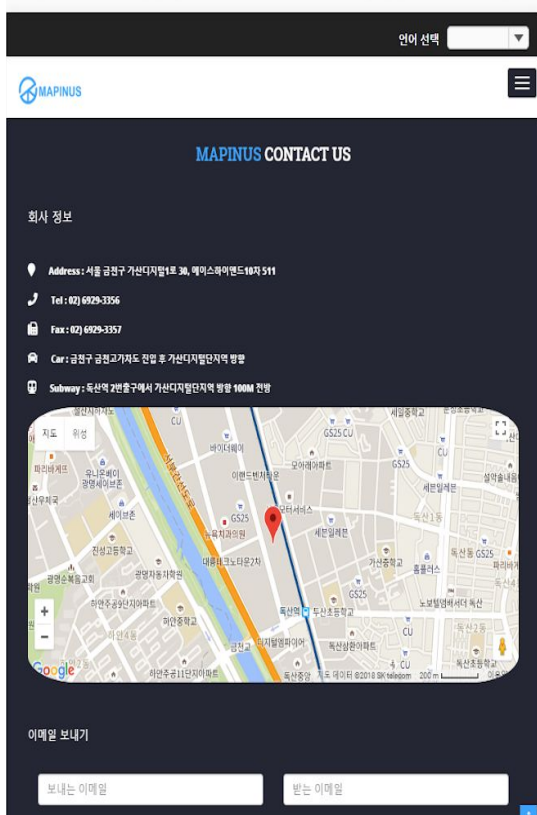
그림 15 특허 및 인증서 확대보기

그림 15는 그림 14의 자세히 보기 버튼을 눌렀을시에 화면을 캡처한 것입니다. 작은 글씨로 된 인증서를 확대하여 볼 경우가 생길수 있기에 만들었습니다. Fancybox를 이용하였습니다.

참고 URL : <https://github.com/fancyapps/fancybox>

3-3 반응형 서비스





반응형으로 제작한 웹에서는 화면을 줄였을 시에 배치가 깨지지 않습니다. 뿐만 아니라 모바일에서도 화면이 깨지지 않습니다.

3-4 기타

3-4-1 loader



그림 16 Loader 화면

그림 16은 로딩화면을 캡처한 것입니다. Fakeloader 와 회사 로고를 사용하여 로딩화면을 제작하였습니다.

참고 URL : <https://github.com/jimkeller/fakeloader>

3-4-2 Favicon

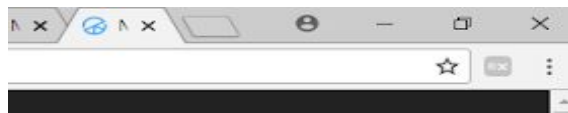


그림 17 Favicon

그림 17은 웹사이트의 탭쪽을 확대한 화면입니다. 회사 로고를 이용하여 Favicon을 작성하여 탭에 로고가 표시 되도록 하였습니다.