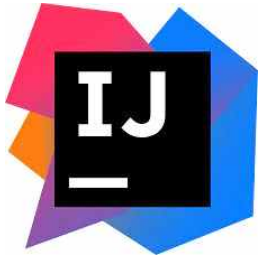
목차

기술 스택 소개

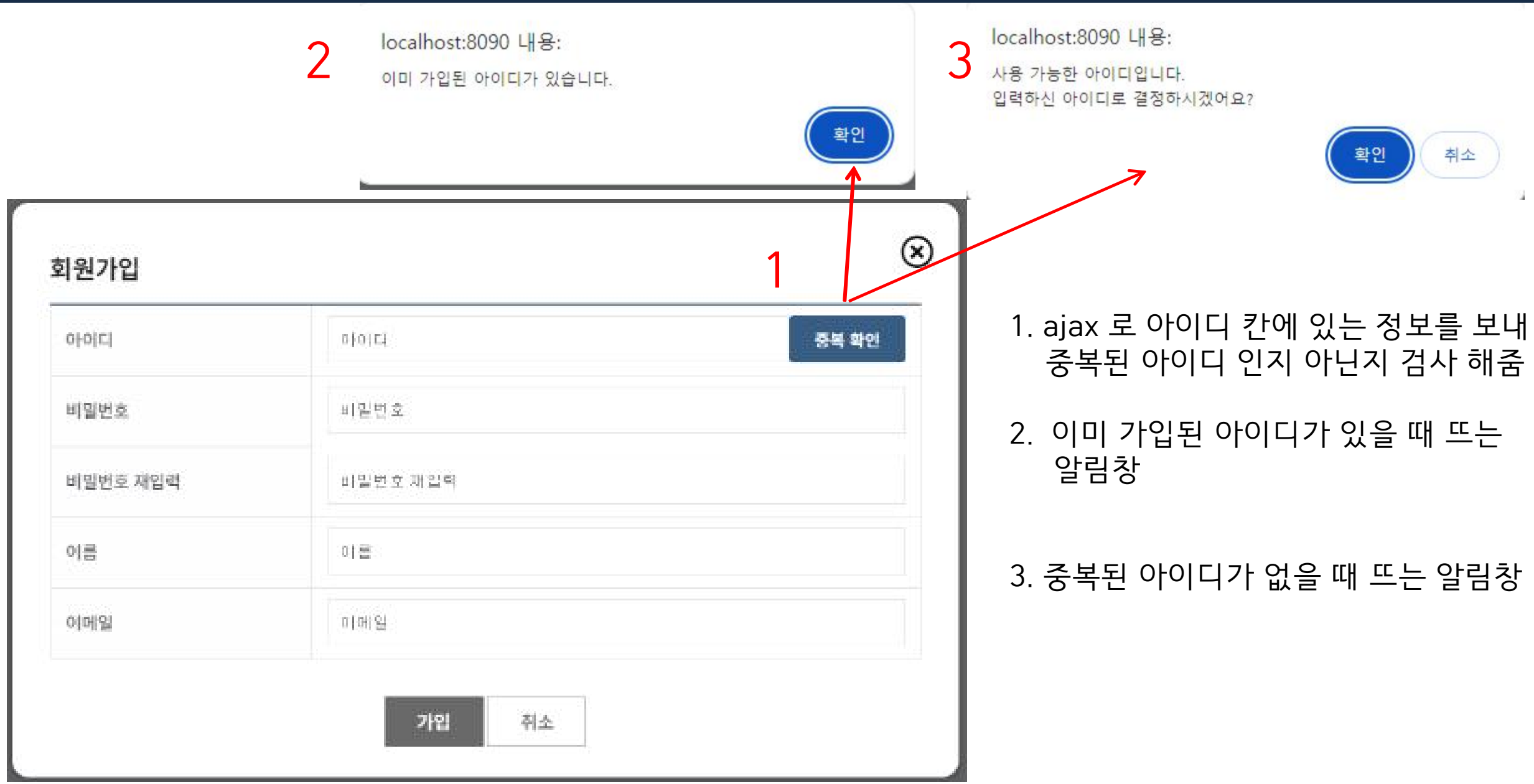


로그인 , 회원가입

1

1. 초기 화면으로 로그인과 회원가입 기능이 있음

로그인 , 회원가입



1. ajax 로 아이디 칸에 있는 정보를 보내 중복된 아이디 인지 아닌지 검사 해줌

2. 이미 가입된 아이디가 있을 때 뜨는 알림창

3. 중복된 아이디가 없을 때 뜨는 알림창

로그인 , 회원가입

2

1

ARCHIVIST

ARMS for SaaS

아키비스트 I 클라우드 기반 차세대 기록관리 시스템

1. 회원 가입 후 아이디와 비밀번호를 치면나오게 되는 메인화면

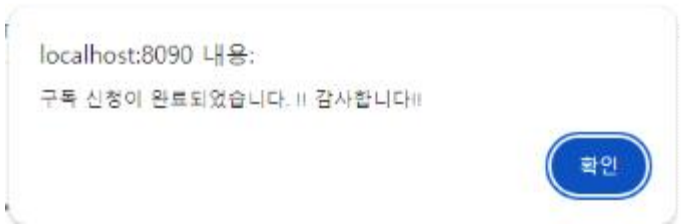
2. 왼쪽 레이 아웃 버튼을 클릭하면 나오는 기능들

구독 신청, 결제, 기간 연장

1

2

3



4

1. 메인 화면에서 구독 신청을 클릭시 나오게 되는 화면, 필수 정보와 부가 정보를 입력한 뒤 구독 신청이 가능 하다.

2. 구독 신청 버튼을 누르면 카카오 페이 화면으로 이동 하게 된다.

3. 카카오 페이에서 테스트 결제를 완료하게 되면 메인 페이지로 이동하게 되고 뜨는 알림창

4. 구독한 뒤로는 레이아웃 기능이 그에 맞게 변한다.

구독 신청, 결제, 기간 연장

1

서비스 등급 : Basic

남은 기간 : 4달

사용 인원 : 4명

회사 이름 : 더 큐피트

이용 가능한 용량 : 4TB

5 구독 취소하기

2

구독 기간 (달)

4

기간 연장 신청

3

localhost:8090 내용:

구독이 연장되었습니다.! 감사합니다!!

확인

4

서비스 등급 : Basic

남은 기간 : 8달

사용 인원 : 4명

회사 이름 : 더 큐피트

이용 가능한 용량 : 4TB

1. 메인에서 현재 구독 정보 클릭시 나오는 화면, 자신이 구독했던 정보를 볼 수 있다.

2. 메인에서 구독 기간 연장 클릭시 나오게 되는 화면 , 원하는 기간을 입력하고 신청 버튼을 누르면 테스트 결제를 하게 된다.

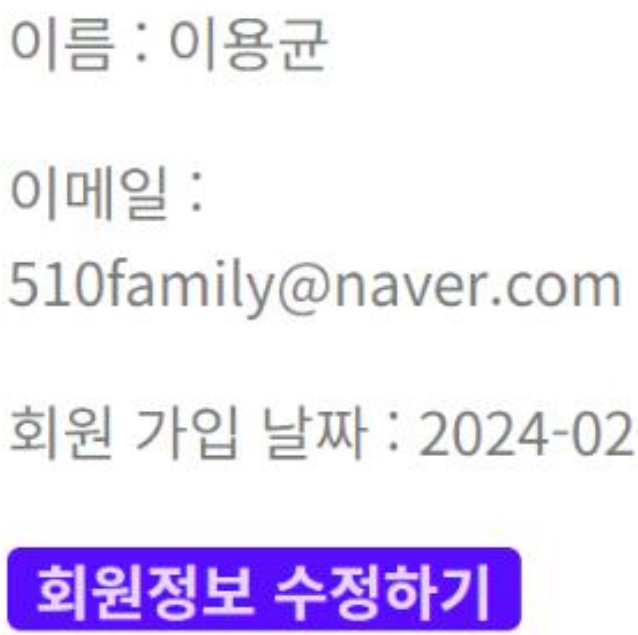
3. 결제가 끝난후 메인으로 오게 되면 뜨는 알림창

4. 기간이 연장된후 정보가 업데이트된 현재 구독 정보 페이지

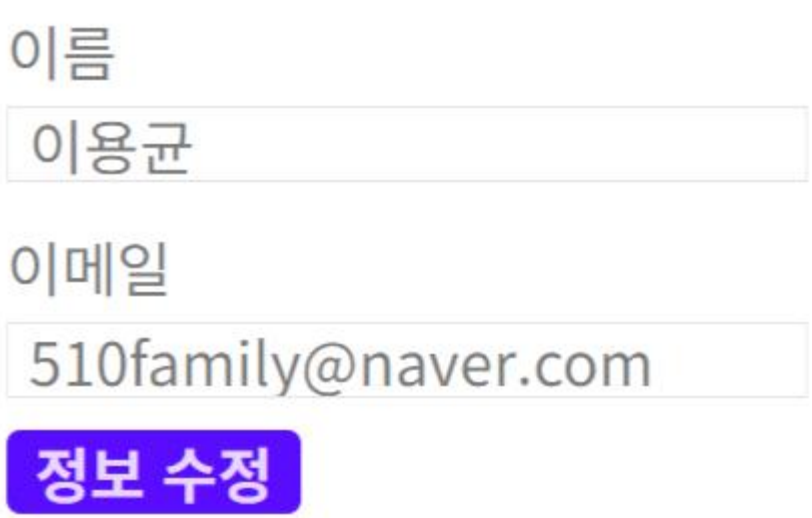
5. 구독 취소하기를 누르면 예 아니요로 한번 더 확인 후 예를 누르면 구독 서비스가 취소가 된다.

마이페이지

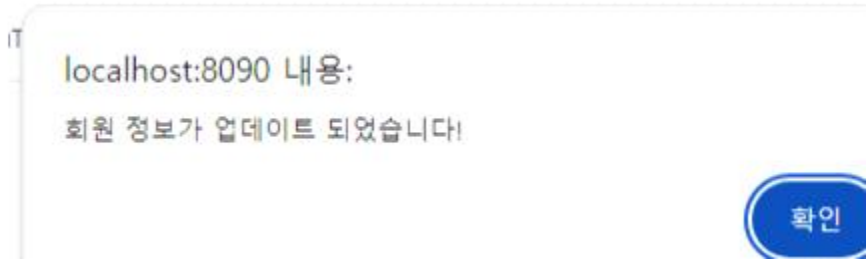
1



2



3



1. 마이페이지 페이지에서 볼수 있는 유저의 정보들

2. 마이페이지에서 회원정부 수정하기 버튼 클릭시 나오는 페이지
   이름과 이메일을 수정할 수 있다.

3. 정보 수정 완료시 뜨는 알림창

REST API 구조 활용

1

```
function callApi(uri, method, params) {
    var token = $("meta[name='_csrf']").attr("content");
    var header = $("meta[name='_csrf_header']").attr("content");
    let json = {};
     JSON.stringify(params);
     $.ajax({
        url : uri,
        type : method,
           xhrFields: {
                withCredentials: true
            },
        contentType : 'application/json; charset=utf-8',
        dataType : 'json',
        data : (params) ? JSON.stringify(params) : {},
        beforeSend : function(xhr)
                    {   /*데이터를 전송하기 전에 헤더에 csrf값을 설정한다*/
                        xhr.setRequestHeader(header, token);
                    },
        async : false,
        success : function (response) {
            json = response;
        },
        error : function (request, status, error) {
            console.log(error)
        }
    })
    console.log(json);
```

2

```
callApi('/user/information/'+userId, 'patch', params);
alert('회원 정보가 업데이트 되었습니다!');
```

3

```
// 회원 정보 수정
no usages   유iguenguen37
@PatchMapping("/information/{id}")
@ResponseBody
public Long updateMember(@PathVariable(name = "id") final Long id, @RequestBody final SiteUserRequest params) {
    System.out.println("회원 정보 수정 시작!!!!!!!!!!!!!!");
    System.out.println(params.getUserName());
    params.setId(id);
    System.out.println(params.getId());
    System.out.println(params.getEmail());
    return userService.updateUser(params);
}
```

1. js 에서 서버로 REST API 구조로 요청하는 함수
2. 프로젝트에서 유저 정보를 수정할때 보내는 코드
3. 서버에서 요청 받으면 실행되는 함수

프로젝트 후기

REST API, 카카오페이 결제, MyBatis 등 처음 도입하는 것들이 많이 있었다.

원래 JPA 로만 프로젝트를 만들어봤어서 MyBatis 도 첫 시도 였다.
그래서 테스트 과제를 받고 쉽지 않겠다라는 느낌이 있었지만
주저하지 않고 배우면서 만들었다.

그렇게 프로젝트를 완성하면서 점점 감을 잡게 되고 완성후에는
많은 경험과 정보들을 얻게 되었다.

특히 REST API 구조로 만들게되면 더 범용성 있고 시각화된 코드를 만들
있다는 것을 깨달았다.

참고한 정보

- https://lealea.tistory.com/118

- https://congsong.tistory.com/35