

2024년 상반기 K-디지털 트레이닝

# 회원관리-비밀번호 변경(백엔드)

[KB] IT's Your Life

## 회원관리-비밀번호 변경(백엔드)

- ✓ 비밀번호 변경에는 multipart가 없으므로, application/json 인코딩으로 처리
  - DTO 앞에 @RequestBody 설정 필요!

### ChangePasswordDTO.java

```
@Data
@NoArgsConstructor
@AllArgsConstructor
public class ChangePasswordDTO {
    String username;    // 사용자 ID
    String oldPassword; // 이전 비밀번호
    String newPassword; // 새 비밀번호
}
```

### MemberMapper.java

```
public interface MemberMapper {  
    ...  
    int updatePassword(ChangePasswordDTO changePasswordDTO);  
}
```

### MemberMapper.xml

```
<update id="updatePassword">
  UPDATE tbl_member
  SET
    password = #{newPassword},
    update_date = now()
  WHERE username =#{username}
</update>
```

### MemberService.java

```
public interface MemberService {  
    ...  
  
    void changePassword(ChangePasswordDTO changePassword);  
}
```

### MemberServiceImpl.java

```
@Override
public void changePassword(ChangePasswordDTO changePassword) {
    MemberVO member = mapper.get(changePassword.getUsername());

    if(!passwordEncoder.matches(changePassword.getOldPassword(), member.getPassword())) {
        throw new PasswordMismatchException();
    }

    changePassword.setNewPassword(passwordEncoder.encode(changePassword.getNewPassword()));

    mapper.updatePassword(changePassword);
}
```

### MemberController.java

```
@PostMapping("/{username}/changepassword")
public ResponseEntity<?> changePassword(@RequestBody ChangePasswordDTO changePasswordDTO) {
    service.changePassword(changePasswordDTO);
    return ResponseEntity.ok().build();
}
```





2024년 상반기 K-디지털 트레이닝


# 회원관리-비밀번호 변경(프론트엔드)

[KB] IT's Your Life

### 비밀번호 변경

 이전 비밀번호:

 새 비밀번호:

 새 비밀번호 확인:

✓ 확인

## api/authApi.js

```
import api from '@api';

const BASE_URL = '/api/member';

export default {
  ...
  async changePassword(formData) {
    const { data } = await api.put(`${BASE_URL}/${formData.username}/changepassword`, formData);
    console.log('AUTH PUT: ', data);

    return data;
  },
};
```

## pages/auth/ChangePasswordPage.vue

```
<script setup>
import { computed, reactive, ref } from 'vue';
import { useAuthStore } from '@stores/auth';
import { useRouter } from 'vue-router';
import authApi from '@api/authApi';

const router = useRouter();
const auth = useAuthStore();

const changePassword = reactive({
  username: auth.username,
  oldPassword: '',
  newPassword: '',
  newPassword2: '',
});

const disableSubmit = computed(() => !changePassword.oldPassword || !changePassword.newPassword
  || !changePassword.newPassword2);

const error = ref('');
```

## pages/auth/ChangePasswordPage.vue

```
const inputPassword = () => (error.value = '');

const resetError = () => (error.value = '');

const onSubmit = async () => {
  if (changePassword.newPassword !== changePassword.newPassword2) {
    error.value = '새 비밀번호가 일치하지 않습니다.';
    return;
  }

  try {
    await authApi.changePassword(changePassword);
    alert('비밀번호를 수정했습니다.');
    router.push({ name: 'profile' });
  } catch (e) {
    error.value = e.response.data;
  }
};
</script>
```

## pages/auth/ChangePasswordPage.vue

```
<template>
  <div class="mt-5 mx-auto" style="width: 500px">
    <h1 class="my-5">
      <i class="fa-solid fa-lock"></i>
      비밀번호 변경
    </h1>

    <form @submit.prevent="onSubmit">
      <div class="mb-3">
        <label for="password" class="form-label">
          <i class="fa-solid fa-lock"></i>
          이전 비밀번호:
        </label>
        <input type="password" class="form-control" placeholder="이전 비밀번호" v-model="changePassword.oldPassword"
          @input="inputPassword" />
      </div>
```

## pages/auth/ChangePasswordPage.vue

```
<div class="mb-3">
  <label for="password" class="form-label">
    <i class="fa-solid fa-lock"></i>
    새 비밀번호:
  </label>
  <input type="password" class="form-control" placeholder="새 비밀번호" v-model="changePassword.newPassword"
    @input="resetError" />
</div>

<div class="mb-3">
  <label for="password" class="form-label">
    <i class="fa-solid fa-lock"></i>
    새 비밀번호 확인:
  </label>
  <input type="password" class="form-control" placeholder="새 비밀번호 확인" v-model="changePassword.newPassword2"
    @input="resetError" />
</div>

<div v-if="error" class="text-danger">{{ error }}</div>
```

### pages/auth/ChangePasswordPage.vue

```
<button type="submit" class="btn btn-primary mt-4" :disabled="disableSubmit">
  <i class="fa-solid fa-check"></i>
  확인
</button>
</form>
</div>
</template>
```



## ✓ Spring Security 설정

### ○ 인증 요구 경로

- POST :: /api/member
- PUT :: /api/member
- PUT :: /api/member/\*/changepassword

## SecurityConfig.java

```
@Override
public void configure(HttpSecurity http) throws Exception {
    ...
    http
        .authorizeRequests()
        .antMatchers(HttpMethod.OPTIONS).permitAll()
        .antMatchers(HttpMethod.POST, "/api/member").authenticated()
        .antMatchers(HttpMethod.PUT, "/api/member", "/api/member/*/changepassword").authenticated()
        .anyRequest().permitAll();

    http.httpBasic().disable() // 기본 HTTP 인증 비활성화
        .csrf().disable() // CSRF 비활성화
        .formLogin().disable() // formLogin 비활성화 관련 필터 해제
        .sessionManagement().sessionCreationPolicy(SessionCreationPolicy.STATELESS); // 세션 생성 모드 설정
}
```