

## SpringBoot Email Service Password Change UI

### Add this in application.properties

```
# Site domain - base url - email password settings
site.domain = http://localhost:8080/
```

### Add this in AccountController.java

```
@Value("${site.domain}")
private String site_domain;
```

### Update From below code

```
String reset_message = "This is the reset password link : http://localhost/reset-
password?token"+ reset_token;
```

### To

```
String reset_message = "This is the reset password link : "+site_domain+"change
password?token ="+ reset_token;
```

### Adding endpoint for change password in AccountController.java – Email

```
// Token is the parameter which is present in the URL, that was sent when reset
// button clicked and stored in db ; ("token")
@GetMapping("/change-password")
public String change_password(Model model, @RequestParam("token") String
token, RedirectAttributes attributes) {
    Optional<Account> optional_account = accountService.findByToken(token);
    if (optional_account.isPresent()) {
        long account_id = optional_account.get().getId();
        LocalDateTime now = LocalDateTime.now();
        if
(now.isAfter(optional_account.get().getPassword_reset_token_expiry())) {
            attributes.addFlashAttribute("error", "Token expired");
            return "redirect:/forgot-password";
        }
        model.addAttribute("account_id", account_id);
        return "account_views/change-password";
    }
    attributes.addFlashAttribute("error", "Invalid token");
    return "redirect:/forgot-password";
}
```

## In Account model -> modify the property -> password\_reset\_token

```
@Column(name = "token")
private String password_reset_token;
```

Change in Column Name

ID	AGE	DATE_OF_BIRTH	EMAIL	FIRSTNAME	GENDER	LASTNAME	PASSWORD	TOKEN	PASSWORD_RESET_TOKEN_EXPIRY	PHOTO
1	25	1990-01-01	user@user.com	User	Male	lastname	\$2a\$10\$TIOGEPmVQYzRo3trw17z1C05Cma.E5vNcYUxSSmReVMH2QntH.0K7...	null	null	/resources/static/images/person.png
2	25	1990-01-01	admin@studyeasy.org	Admin	Male	lastname	\$2a\$10\$zK8rnnjKgrZkzyoF0iOux7i6SiasQrPkoOueDmVvsN8huDnS...	null	null	/resources/static/images/person.png
3	55	1998-02-01	editor@editor.com	Editor	Female	lastname	\$2a\$10\$bR8zvik7pke.tqNAY04veCej.k3A8.lMcSEX0.eRs/GdA2vJyK...	null	null	/resources/static/images/person.png
4	40	1973-08-01	super_editor@editor.com	Editor	Male	lastname	\$2a\$10\$6SY6UNSDznNgNRZa1iaJyVT9CsRvT8iHSjXL73r2aCPvChVJe...	null	null	/resources/static/images/person.png
5	40	1973-08-01	testemailmj2024@gmail.com	Test	Male	Demo Email send	\$2a\$10\$ScNv87NiroVdxx03cenm.WGguCPQYsZBiZSui2DzmYAqkXP...	null	null	/resources/static/images/person.png

## In AccountRepository -> add this code below

```
public Optional<Account> findByToken(String token) {
    return accountRepository.findByToken(token);
}
```

## In AccountService -> add this code below

```
public Optional<Account> findByToken(String token) {
    return accountRepository.findByToken(token);
}
```

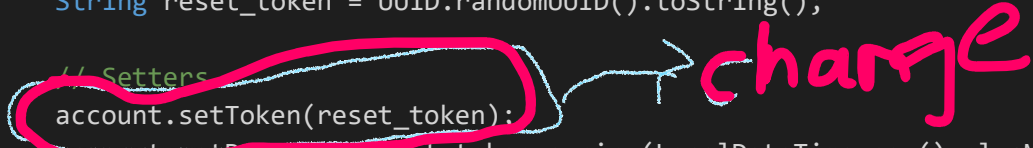
In Account model -> modify the property -> password\_reset\_token

But as we have put the column as token -> change it to token

```
@Column(name = "token")  
private String token;
```

In reset-password.html -> Update like below

```
@PostMapping("/reset-password")  
public String reset_password(@RequestParam("email") String _email,  
RedirectAttributes attributes, Model model) {  
    Optional<Account> optional_account =  
accountService.findOneByEmail(_email);  
    if (optional_account.isPresent()) {  
        Account account =  
accountService.findById(optional_account.get().getId()).get();  
  
        // UUID Token Generation  
        String reset_token = UUID.randomUUID().toString();  
  
        // Setters  
        account.setToken(reset_token);  
        account.setPassword_reset_token_expiry(LocalDateTime.now().plusMinute  
s(password_token_timeout));  
  
        // Saving details  
        accountService.save(account);  
  
        // Email Code  
        String reset_message = "This is the reset password link: " +  
site_domain + "change-password?token=" +  
        + reset_token;  
        emailDetails emailDetails = new emailDetails(account.getEmail(),  
reset_message,  
        "Reset Password Studyeasy demo");  
        if (emailService.sendSimpleEmail(emailDetails) == false) {  
            attributes.addFlashAttribute("error", "Error while sending email,  
contact admin");  
        }  
  
        // Final Flash message  
        attributes.addFlashAttribute("message", "Password reset email sent");  
        return "redirect:/login";  
    }  
}
```



```
    } else {  
        attributes.addFlashAttribute("error", "No user found with the email  
supplied");  
        return "redirect:/forgot-password";  
    }  
}
```

## Output

