

# login-registration backend APIs

👤 Created by	👤 Oskar Shrestha
🕒 Created time	@March 28, 2023 4:39 PM
👤 Last edited by	👤 Oskar Shrestha
🕒 Last edited time	@March 28, 2023 5:03 PM
☰ Tags	Engineering

## GET

### #1 Verify email

API	<code>"/api/v1/auth/verifyEmail"</code>
Request parameter 1	"token" : String
Description	
Response	VerifyEmailResponse

```
public class VerifyEmailResponse {  
    private EmailVerificationTokenStatus emailVerificationTokenStatus;  
}  
public enum EmailVerificationTokenStatus {  
    VALID,  
    INVALID,  
    EXPIRED,  
    DEFAULT,  
}
```

### #2 Resend verify email

API	<code>"/api/v1/auth/resendEmailVerification"</code>
Request parameter 1	"email" : String
Description	
Response	ResendVerifyEmailResponse

```
public class ResendVerifyEmailResponse {
    private ResendVerifyEmailStatus resendVerifyEmailStatus;
}
public enum ResendVerifyEmailStatus {
    SUCCESS,
    EMAIL_NOT_REGISTERED,
}
```

### #3 Forget password

API	<code>"/api/v1/auth/forgetPassword"</code>
Request parameter 1	"email" : String
Description	
Response	ForgetPasswordResponse

```
public class ForgetPasswordResponse {
    private ForgetPasswordEmailStatus forgetPasswordEmailStatus;
}
public enum ForgetPasswordEmailStatus {
    SENT,
    EMAIL_NOT_FOUND,
}
```

### #4 Reset forget password email

API	<code>"/api/v1/auth/resendForgetPasswordEmail"</code>
Request parameter 1	"email" : String
Description	
Response	ResendForgetPasswordEmailResponse

```
public class ResendForgetPasswordEmailResponse {
    private ResendForgetPasswordEmailStatus resendForgetPasswordEmailStatus;
}
public enum ResendForgetPasswordEmailStatus {
    SENT,
    EMAIL_NOT_REGISTERED,
}
```

## POST

## #1 Register user

API	<code>"/api/v1/auth/register"</code>
Request body	<code>UserRegisterRequest</code>
Description	
Response	<code>RegistrationResponse</code>

```
public class UserRegisterRequest {  
    private String firstName;  
    private String lastName;  
    private String email;  
    private String password;  
}  
public class RegistrationResponse {  
    private boolean existingUser;  
  
    private boolean registrationSuccess;  
}
```

## #2 Authenticate user

API	<code>"/api/v1/auth/authenticate"</code>
Request body	<code>UserAuthenticationRequest</code>
Description	
Response	<code>AuthenticationResponse</code>

```
public class UserAuthenticationRequest {  
    private String email;  
    private String password;  
}  
public class AuthenticationResponse {  
    private String token;  
}
```

## #3 Change password

API	<code>"/api/v1/auth/changePassword"</code>
Request body	<code>ChangePasswordRequest</code>
Description	
Response	<code>ChangePasswordResponse</code>

```

public class ChangePasswordRequest {
    private String email;
    private String oldPassword;
    private String newPassword;
}
public class ChangePasswordResponse {
    private ChangeUserPasswordStatus changeUserPasswordStatus;
}
public enum ChangeUserPasswordStatus {
    SUCCESS,
    PASSWORD_MISMATCH,
    EMAIL_NOT_FOUND,
}

```

## #4 Reset password

API	<code>"/api/v1/auth/resetPassword"</code>
Request parameter 1	<code>"token" : String</code>
Request body	<code>ResetPasswordRequest</code>
Description	
Response	<code>ResetPasswordResponse</code>

```

public class ResetPasswordRequest {
    private String password;
}
public class ResetPasswordResponse {
    private ResetPasswordResponseStatus resetPasswordResponseStatus;
}
public enum ResetPasswordResponseStatus {
    SUCCESS,
    TOKEN_NOT_FOUND,
    EXPIRED,
    INVALID,
}

```