

BYPASS EMAIL VERIFICATION API TUTORIAL

Composed by: Charlie Tan

Local Route: http://localhost:5000/secret_api/bypass_email_verification
(secret_api/bypass_email_verification)

Deployed Route: https://kindling-lp.herokuapp.com/secret_api/bypass_email_verification
(secret_api/bypass_email_verification)

IMPORTANT NOTE:

-This API endpoint should only be used for development/testing purposes. Our applications should NOT be relying on this to verify an account. This API endpoint is useful should you need to verify a dummy test account you've created. In this case, the phony email you have provided would be impossible to send a verification email to and thus, you can use this API to bypass this requirement.

-To use this API without incorporating it into our actual applications, just use ARC/Postman/SwaggerInspector/etc to send a POST request to the appropriate route.

EXPECTED INPUT FROM FRONTEND:

```
{ "email_str" : some_string }
```

INPUT PROPERTIES EXPLAINED:

1) email_str: the email string corresponding to the user we wish to verify without dealing with sending an email with a verification code.

EXPECTED OUTPUT FROM BACKEND:

```
{ "success_bool" : some_boolean }
```

OUTPUT PROPERTIES EXPLAINED:

1) success_bool: whether or not the verification process was successful. If 'true', the user is verified, 'false' otherwise.

EXPECTED OUTPUT ILLUSTRATED:

1)

-Case: client has a 'ready_status' code of zero before verification.

-Expected output:

```
{ "success_bool" : true }
```

-In the above case, the client's 'ready_status' code is set from zero to 1 and is considered verified.

2)

-Case: somehow, the client has a 'ready_status' code of some integer greater than zero before verification.

-Expected output:

```
{ "success_bool" : true }
```

-In the above case, the client's 'ready_status' code is left alone as it is. For example, if their 'ready_status' code was already 2 before verification, the API does nothing and leaves the 'ready_status' code as 2. The client is considered verified even though nothing was done since they had to have been verified in the past to reach a 'ready_status' code of greater than zero.

3)

-Case: a client corresponding to 'email_str' provided by the frontend input does not exist in the database.

-Expected output:

```
{ "success_bool" : false }
```

4)

-Case: a database error occurs somehow and verification can not proceed.

-Expected output:

```
{ "success_bool" : false }
```
