

# Final Project API Documentation

---

For this project I am using multiple user account option, using Google+ for their OAuth source.

Then I use the email address returned via OAuth, since it unique to create a user entity with their email address as the key. Then with each request I check their token, and ensure that their token is valid for the email address that they are trying to access via Google+.

The only thing my API really does is be a middle man between a Google+ API. Then carries on with CRUD requests after validation success.

**Base Url:** `http://35.169.224.183:3009`

---

## User Routes

---

**Route:** `/user`

**Method:** `GET`

### Params

- *Required [Query]:* token
  - Your Google Oauth token to validate against the email of profile
- *Required [Query]:* email
  - The email address of the Google+ account & your account you are trying to specify.

### Description

Check to see if the user is actually authorized to get profile info in & out of DB. Checks their token against Google and if authorized returns their profile.

**Route:** `/user/check`

**Method:** `POST`

### Params

- *Required [Query]: token*
  - Your Google Oauth token to validate against the email of profile specify.

#### Description

Uses the oAuth token to check to see if the user has already made a local account for the project. If they have, it simply validates the token, and returns their data from the DB. If not, it still checks the token, and creates an empty profile for the user to add to.

**Route:** `/user`

**Method:** `DELETE`

#### Params

- *Required [Query]: token*
  - Your Google Oauth token to validate against the email of profile specify.
- *Required [Query]: email*
  - The email address of the Google+ account & your local account you are trying to specify.

#### Description

Validates the oAuth token against Google, if successful then proceeds to remove the entity from the DB.

**Route:** `/user/delivery`

**Method:** `POST`

#### Params

- *Required [Query]: token*
  - Your Google Oauth token to validate against the email of profile specify.
- *Required [Query]: email*
  - The email address of the Google+ account & your local account you are trying to specify.

## Description

Validates the oAuth token against Google, if successful then proceeds to remove the entity from the DB.

**Route:** `/user/delivery`

**Method:** `PATCH`

## Params

- *Required [Query]:* token
  - Your Google Oauth token to validate against the email of profile specify.
- *Required [Query]:* email
  - The email address of the Google+ account & your local account you are trying to specify.
- *Required [Query]:* deliveryId
  - The deliveryId of the delivery you are trying to edit
- *Required [Body]:* delivery
  - Example Delivery object

```
let delivery = {  
  address: '12345 Hacker Way',  
  total: '$20.00',  
  tip: '$5',  
  time: new Date()  
}
```

## Description

Validates the oAuth token against Google, if successful then proceeds to edit the delivery for the user in the DB .

**Route:** `/user/delivery`

**Method:** `DELETE`

## Params

- *Required [Query]:* token

- Your Google Oauth token to validate against the email of profile specify.
- *Required [Query]: email*
  - The email address of the Google+ account & your local account you are trying to specify.
- *Required [Query]: deliveryId*
  - The deliveryId of the delivery you are trying to delete

**Description**

Validates the oAuth token against Google, if successful then proceeds to remove the delivery specified for the user in the DB .