

# LOST AND FOUND FLASK API

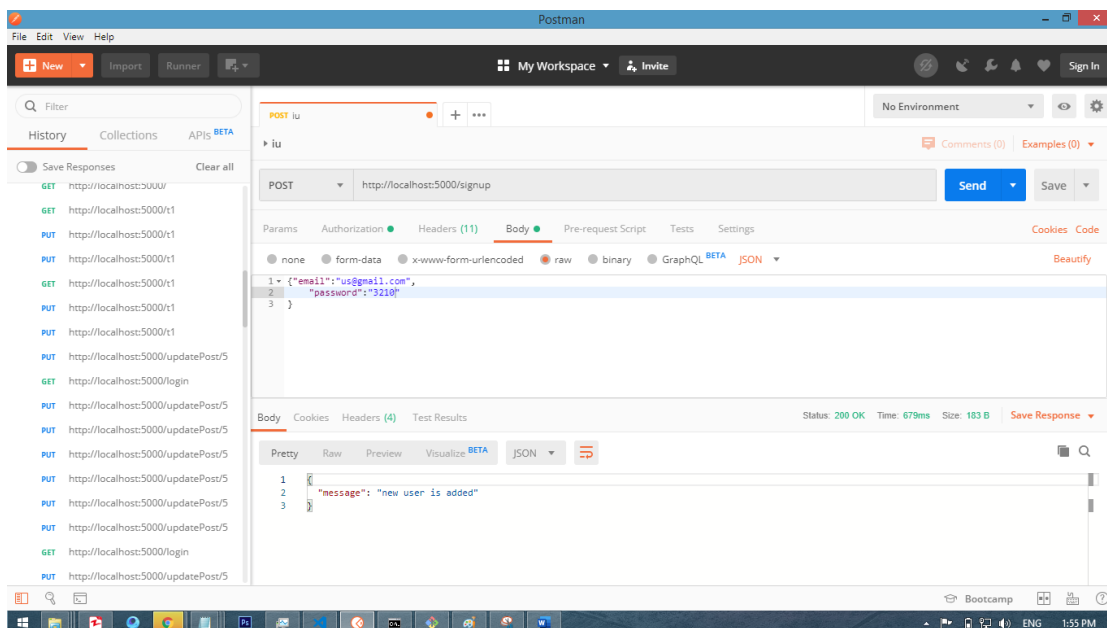
## 1. Authentication

### 1.1 Signup : End user can signup using email and password.

Email is validated for errors, and only unique emails can be registered. API only accepts json object through a POST method. JSON keys should be as follows, i.e.

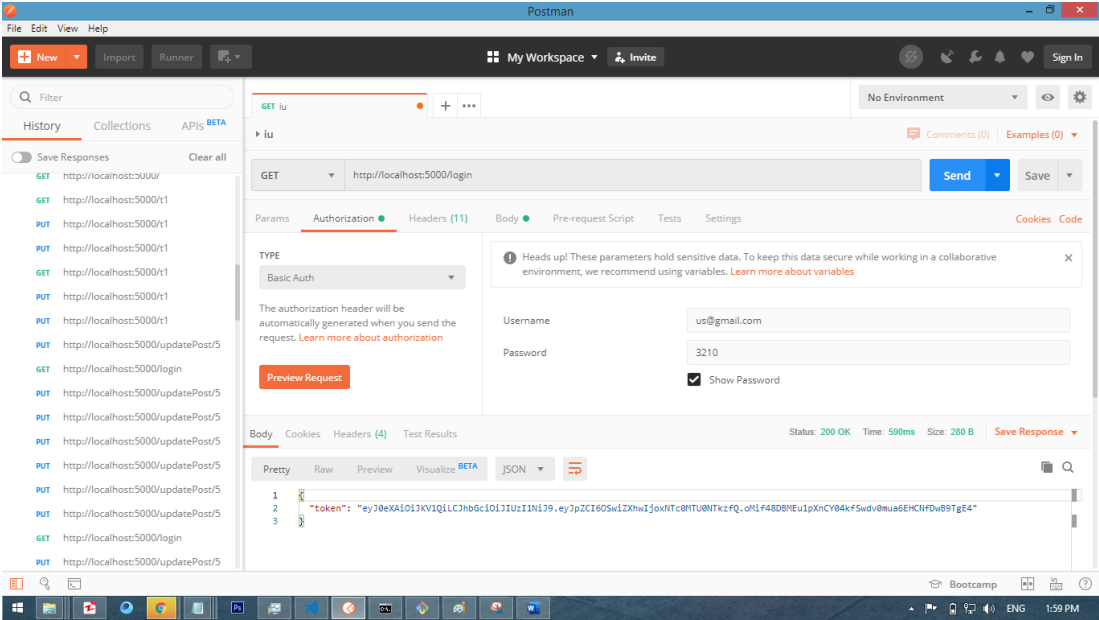
```
{“email”:“example@email.com”,“password”:“somepassword”}
```

**Request URL:** `http://localhost:5000/signup`



### 1.2 Login : Email and password are required for login. On login request, JSON object is created and sent for authentication through GET method. API authenticates the email and password from database and generates a 'access-token' if a user exists. This 'access-token' is used for any further activities.

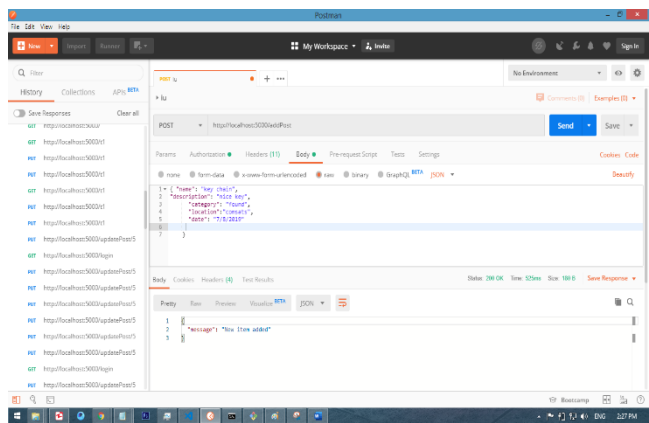
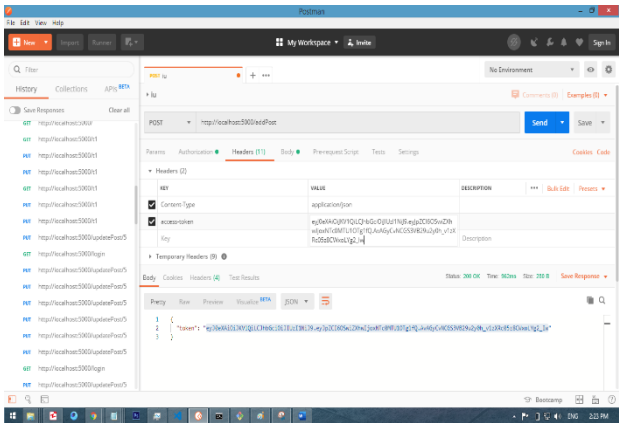
**Request URL:** http://localhost:5000/login



**2. Post Item :** In post item request, an 'access-token' is required in the header of the request. This 'access-token' is verified by the API and it gets a JSON object if it is verified, which is further parsed and inserted into the database. JSON object should be as follows, i.e.

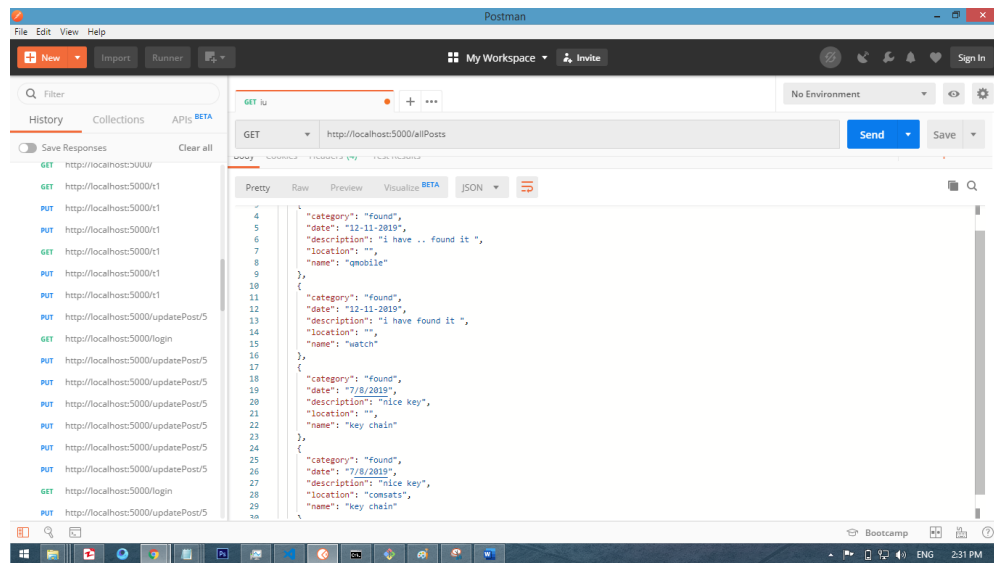
```
{
  "name": "",
  "description": "",
  "category": "lost/found",
  "location": "",
  "date": ""
}
```

**Request URL:** http://localhost:5000/addPost



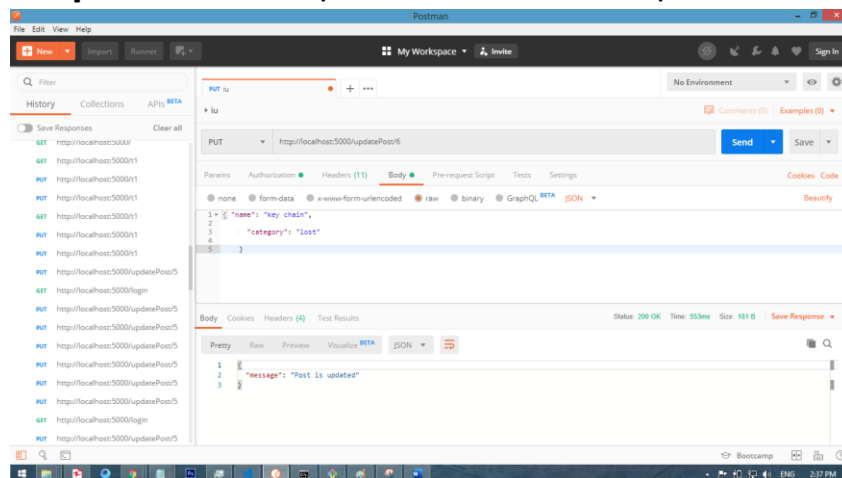
**3. View All Post :** Logged in user can request to view all post using a GET method. API returns a JSON object of all existing items in response.

**Request URL:** `http://localhost:5000/allPosts`



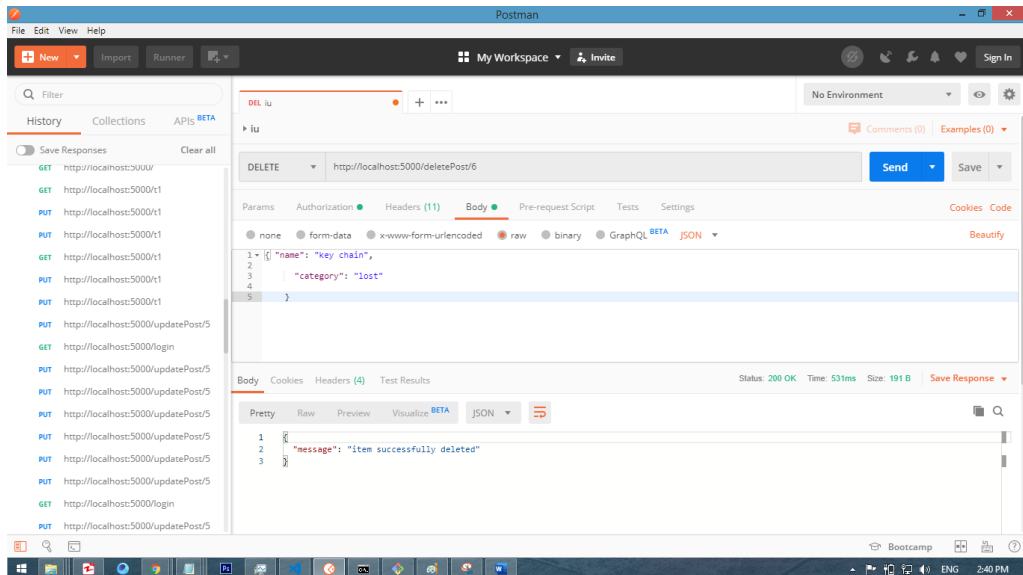
**4. Update Post info :** To update a post, the user must be logged into the system. User can only update the posts he has created. No other user can update each others post. To update a post, user must know the post ID. A PUT request is generated consisting of a JSON object and a header with 'access-token'.

**Request URL:** `http://localhost:5000/updatePost/6`



**5. Delete Post :** To delete a post, the user must be logged into the system. User can only delete the posts he has created. No other user can delete each others post. To delete a post, a DELETE request is generated with post ID passed in the url.

**Request URL:** `http://localhost:5000/deletePost/6`



**6. Search Post (By name) :** Only logged in users can search a post. Posts can be searched using the post name. A GET request is sent with a name in the url. API return a JSON object if a record exists with such name.

**Request URL :** `http://localhost:5000/searchPost/key chain`

