Simon Junior Boateng
Lab 4
Rest API with Flask for Campus Election System

GITHUB REPOSITORY: https://github.com/Ashesi-Org/54272024_election_rest_API.git

1. **REGISTERING A STUDENT AS A VOTER**
   METHOD: POST
   URL: http://127.0.0.1:5000/register

   POST/register

   REQUEST BODY:

   ```
   {
           "student_id": 78962024,
           "name": "David Osei",
           "email": "david.osei@ashesi.edu.gh",
           "department": "Electrical Engineering",
           "gender": "Male",
           "voter_status": "Inactive",
           "created_at": "2022-03-27T12:00:00Z",
           "updated_at": "2022-03-27T12:00:00Z"
   }
   ```

   RESPONSE BODY:
   a. **If the resource being added does not exist already:**
      Status code: 200(ok)

   ```
   {
    "data": {
       "created_at": "2022-03-27T12:00:00Z",
       "department": "Electrical Engineering",
       "email": "david.osei@ashesi.edu.gh",
       "gender": "Male",
       "name": "David Osei",
       "student_id": 78962024,
       "updated_at": "2022-03-27T12:00:00Z",
       "voter_status": "Inactive"
    },
    "message": "Voter registered successfully",
    "status": "success"
   }
   ```
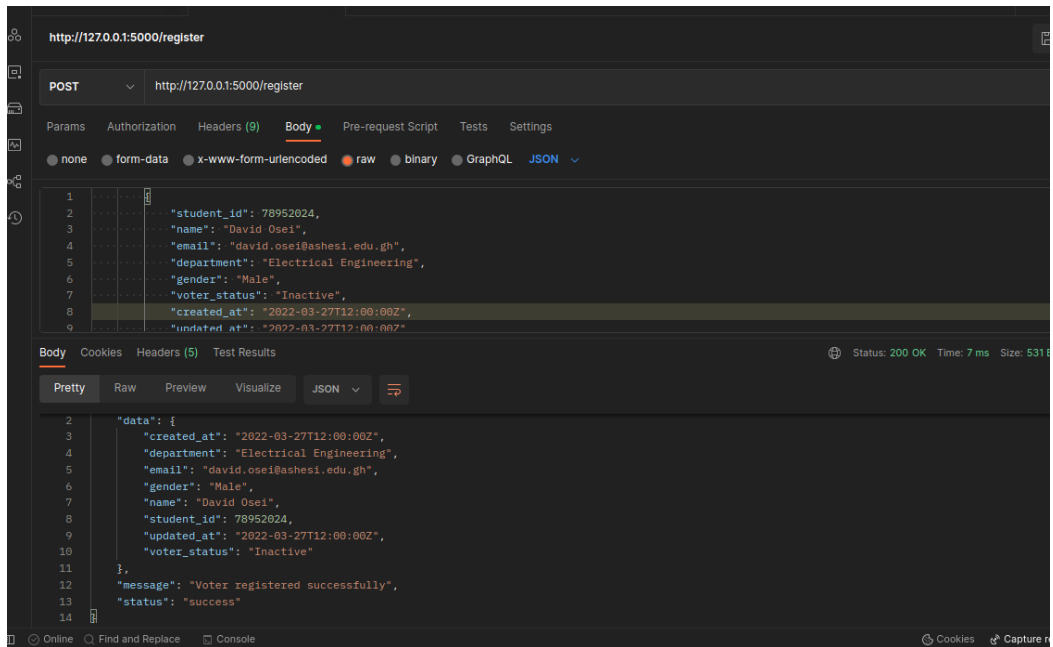
b. If the resource being added does exist already:
Status code: 400(bad request)

```
{
    "message": "This voter already exist",
    "status": "error"
}
```
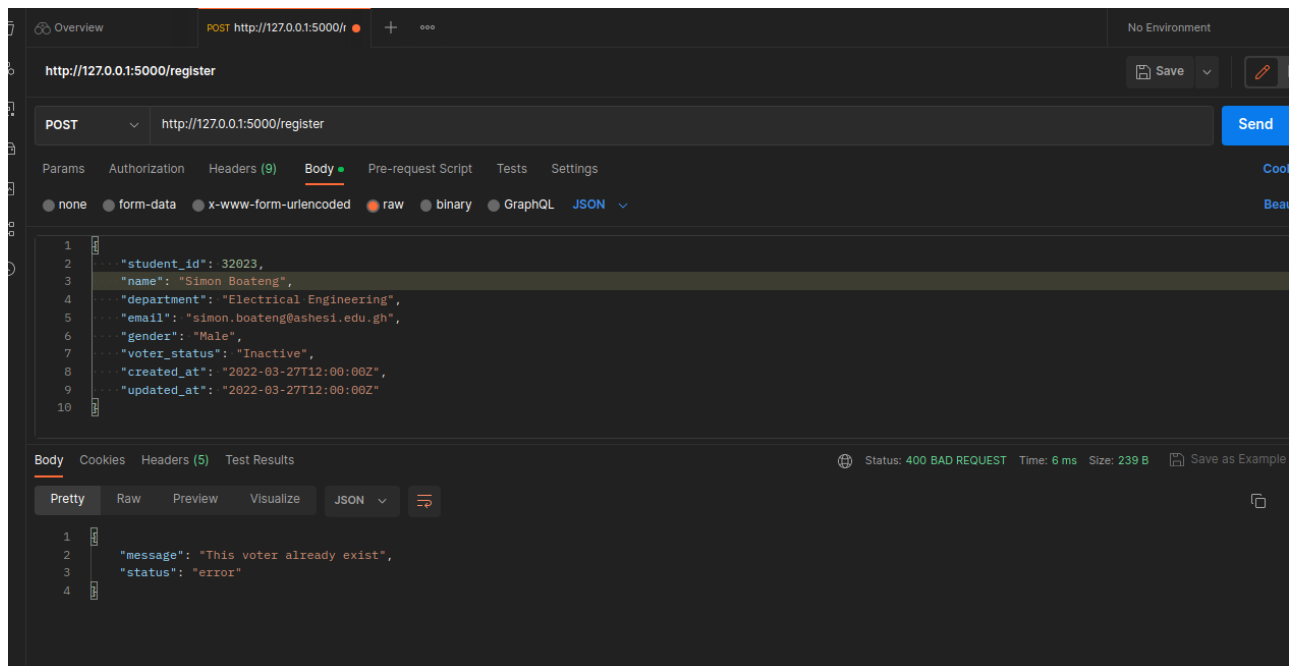
2. REMOVING A VOTER

METHOD: DELETE

URL: http://127.0.0.1:5000/remove_voter/<voter_id>
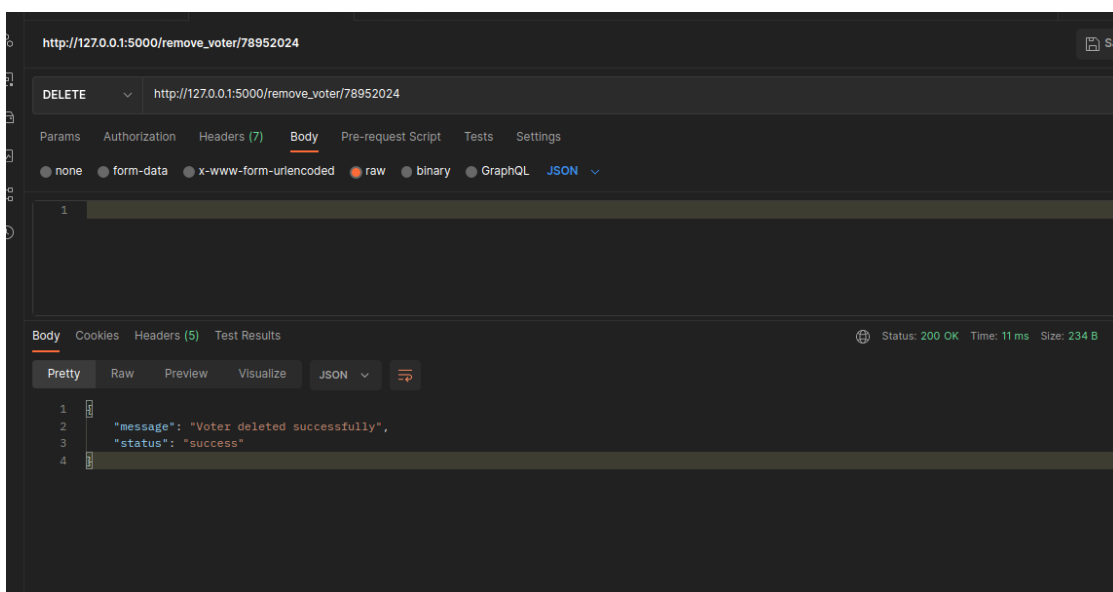
REQUEST: http://127.0.0.1:5000/remove_voter/78952024

RESPONSE BODY:

a. **If the resource being deleted exist:**
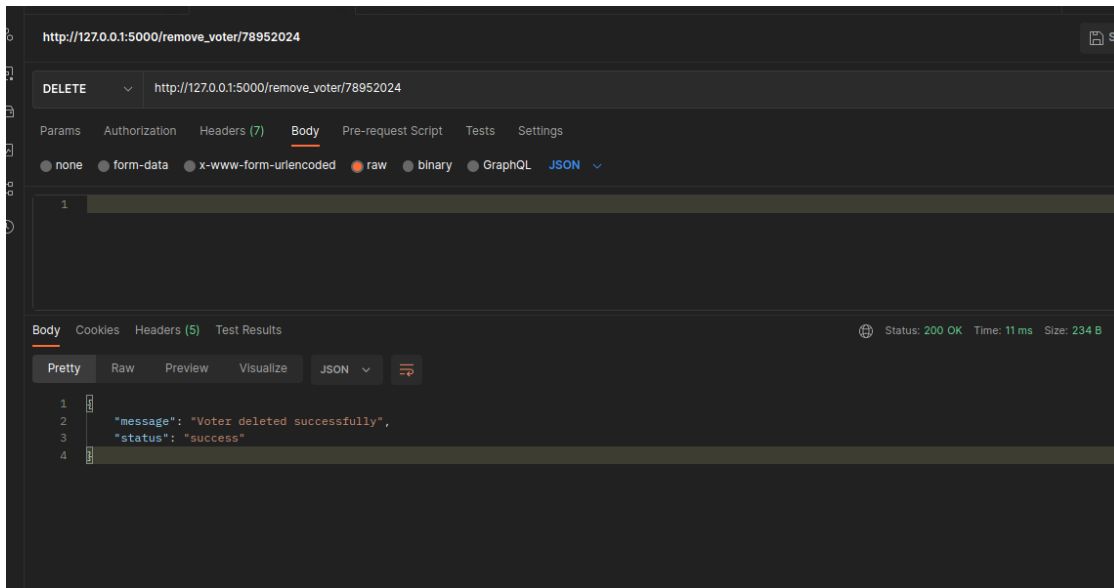
Status code: 200(ok)

```
{
    "message": "Voter deleted successfully",
    "status": "success"
}
```


b. **If the resource being deleted does not exist already:**

Status code: 404

```
{
    "message": "Voter with that id does not exist",
    "status": "not found"
}
```

3. UPDATING VOTER DETAILS
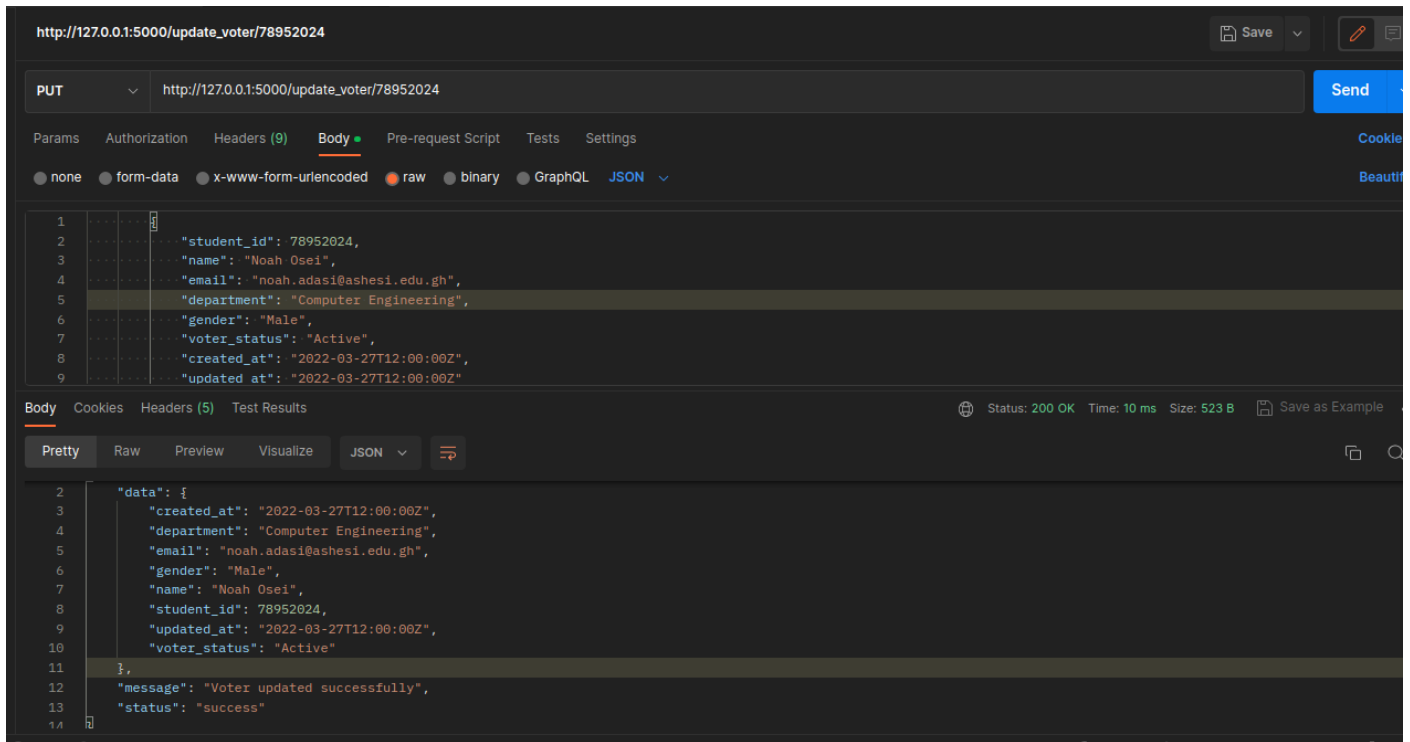
REQUEST
UPDATE/update_voter<int:id>

RESPONSE
    a. **If a voter with the given id exist:**
Status code: 200(ok)

200(ok)

```
{
  "data": {
    "created_at": "2022-03-27T12:00:00Z",
    "department": "Computer Engineering",
    "email": "noah.adasi@ashesi.edu.gh",
    "gender": "Male",
    "name": "Noah Osei",
    "student_id": 78952024,
    "updated_at": "2022-03-27T12:00:00Z",
    "voter_status": "Active"
  },
  "message": "Voter updated successfully",
  "status": "success"
}
```
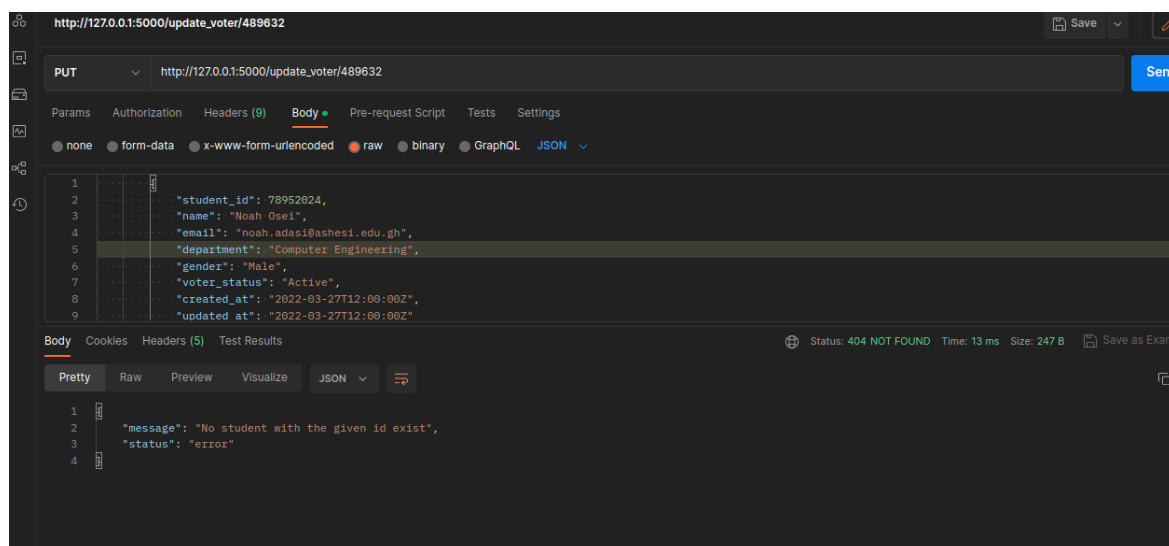
b. **If there is no voter with the given id**
   RESPONSE:
   Status code: 404(Not found)

```
{
    "message": "No student with the given id exist",
    "status": "error"
}
```
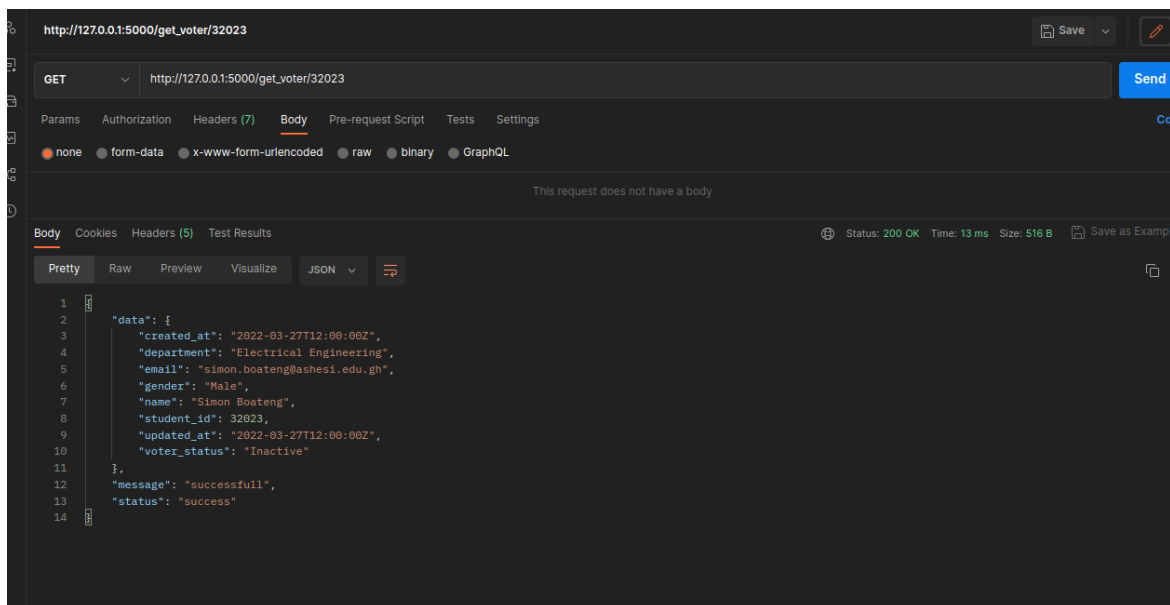
4. RETRIEVING A VOTER BY ID

REQUEST:
GET/get_voter/<int:id>

a. **If the voter exists;**
   REPONSE:
   200(Ok)

```
{
    "data": {
        "created_at": "2022-03-27T12:00:00Z",
        "department": "Electrical Engineering",
        "email": "simon.boateng@ashesi.edu.gh",
        "gender": "Male",
        "name": "Simon Boateng",
        "student_id": 32023,
        "updated_at": "2022-03-27T12:00:00Z",
        "voter_status": "Inactive"
    },
    "message": "successfull",
    "status": "success"
}
```
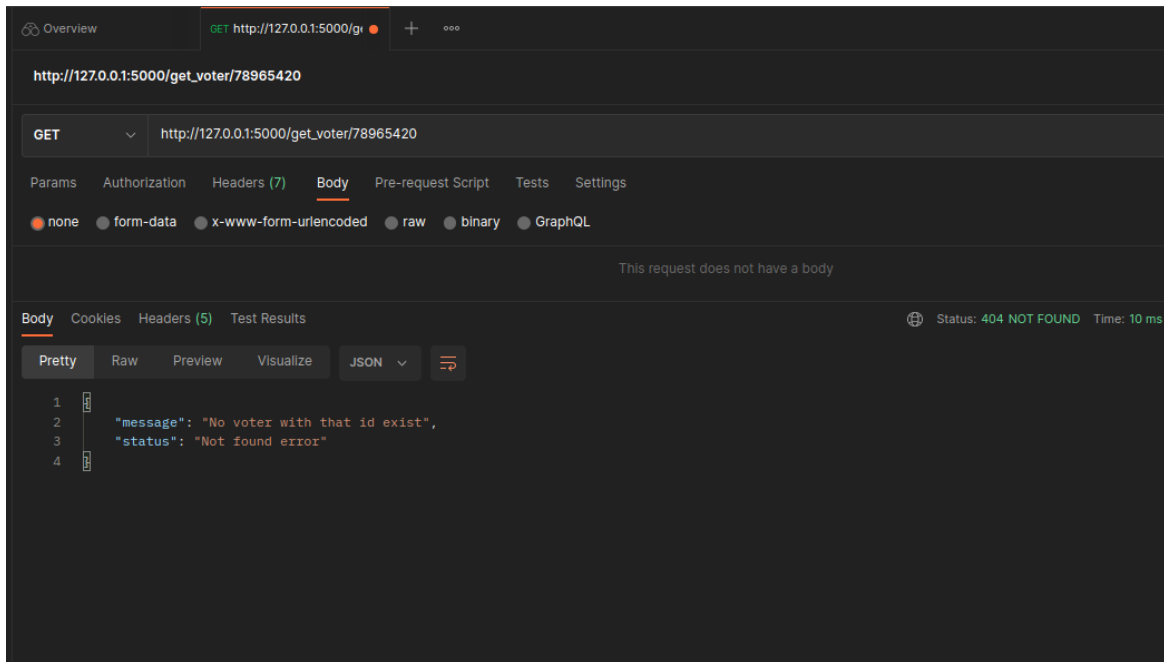

b. **If the voter does not exist**

   RESPONSE:
   404(Not found)

   Response body:

```
{
```

"message": "No voter with that id exist",
    "status": "Not found error"
}

5. CREATING ELECTION

REQUEST:
POST/create_election

REQUEST BODY
```
    {
        "election_id": 1,
        "name": "Student Council Elections",
        "start_time": "2022-04-01T08:00:00Z",
        "end_time": "2022-04-01T18:00:00Z",
        "description": "Election to choose the next student council leaders.",
        "election_type": "General",
        "status": "Scheduled",
        "created_at": "2022-03-27T12:00:00Z",
        "updated_at": "2022-03-27T12:00:00Z"
    }
```
a. If successfully created the election
RESPONSE:
201(created)
```
        {
            "data": {
                "created_at": "2022-03-27T12:00:00Z",
                "description": "Election to choose the next student council leaders.",
                "election_id": 1,
                "election_type": "General",
```
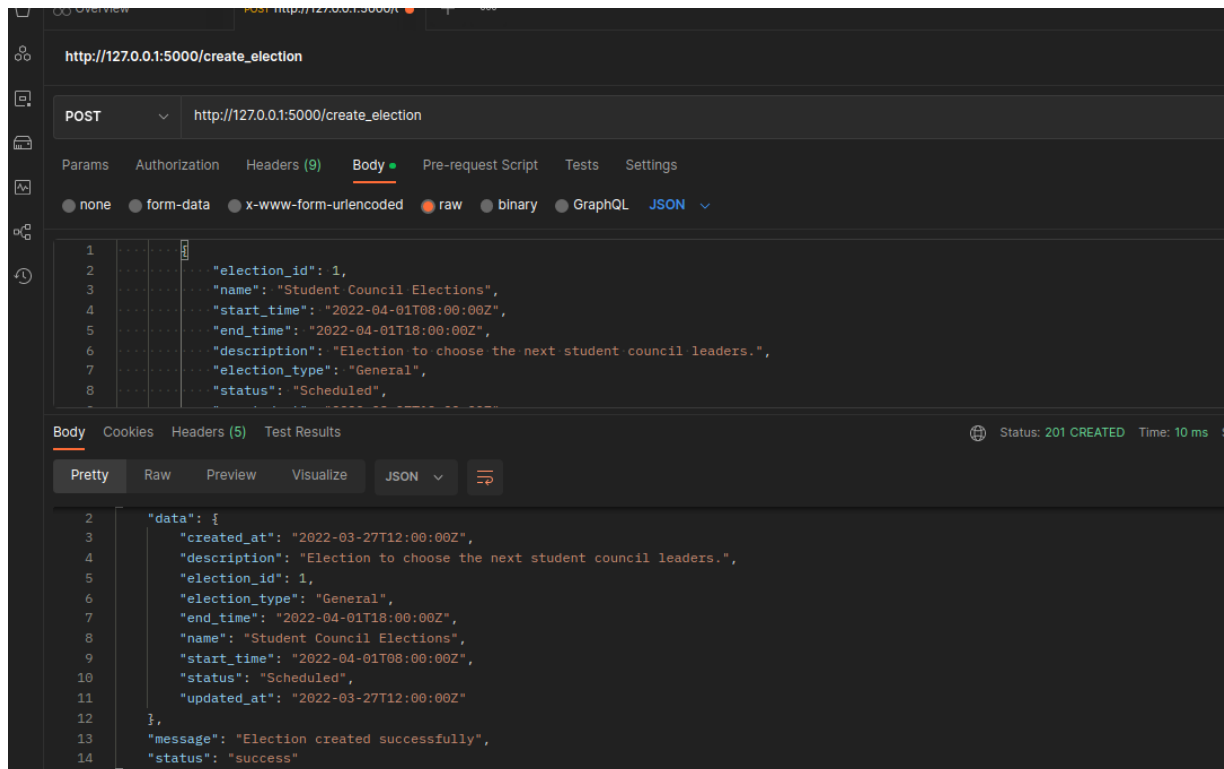
```
        "end_time": "2022-04-01T18:00:00Z",
        "name": "Student Council Elections",
        "start_time": "2022-04-01T08:00:00Z",
        "status": "Scheduled",
        "updated_at": "2022-03-27T12:00:00Z"
    },
    "message": "Election created successfully",
    "status": "success"
}
```

http://127.0.0.1:5000/create_election

POST   http://127.0.0.1:5000/create_election

Params   Authorization   Headers (9)   Body ●   Pre-request Script   Tests   Settings

● none   ● form-data   ● x-www-form-urlencoded   ● raw   ● binary   ● GraphQL   JSON ∨

```
1    {
2        "election_id": 1,
3        "name": "Student Council Elections",
4        "start_time": "2022-04-01T08:00:00Z",
5        "end_time": "2022-04-01T18:00:00Z",
6        "description": "Election to choose the next student council leaders.",
7        "election_type": "General",
8        "status": "Scheduled",
```

Body   Cookies   Headers (5)   Test Results   Status: 201 CREATED   Time: 10 ms

Pretty   Raw   Preview   Visualize   JSON ∨

```
2        "data": {
3            "created_at": "2022-03-27T12:00:00Z",
4            "description": "Election to choose the next student council leaders.",
5            "election_id": 1,
6            "election_type": "General",
7            "end_time": "2022-04-01T18:00:00Z",
8            "name": "Student Council Elections",
9            "start_time": "2022-04-01T08:00:00Z",
10           "status": "Scheduled",
11           "updated_at": "2022-03-27T12:00:00Z"
12       },
13       "message": "Election created successfully",
14       "status": "success"
```
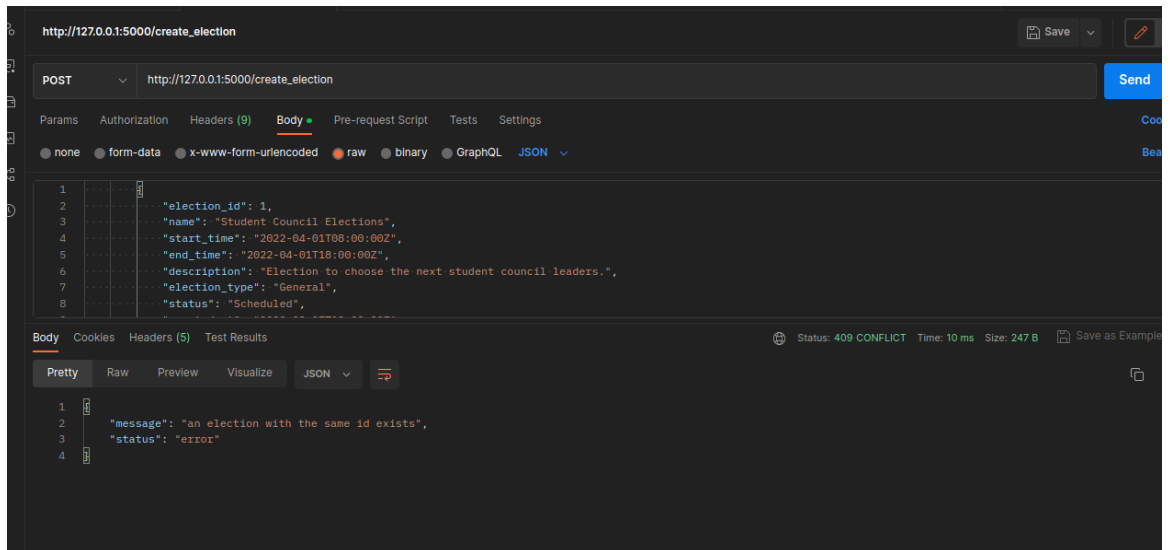
b. **If the election with that id already exists**
   RESPONSE:
   409(conflict)

```
{
    "message": "an election with the same id exists",
    "status": "error"
}
```

6. DELETING AN ELECTION

REQUEST:
DELETE/delete_election/<int:id>
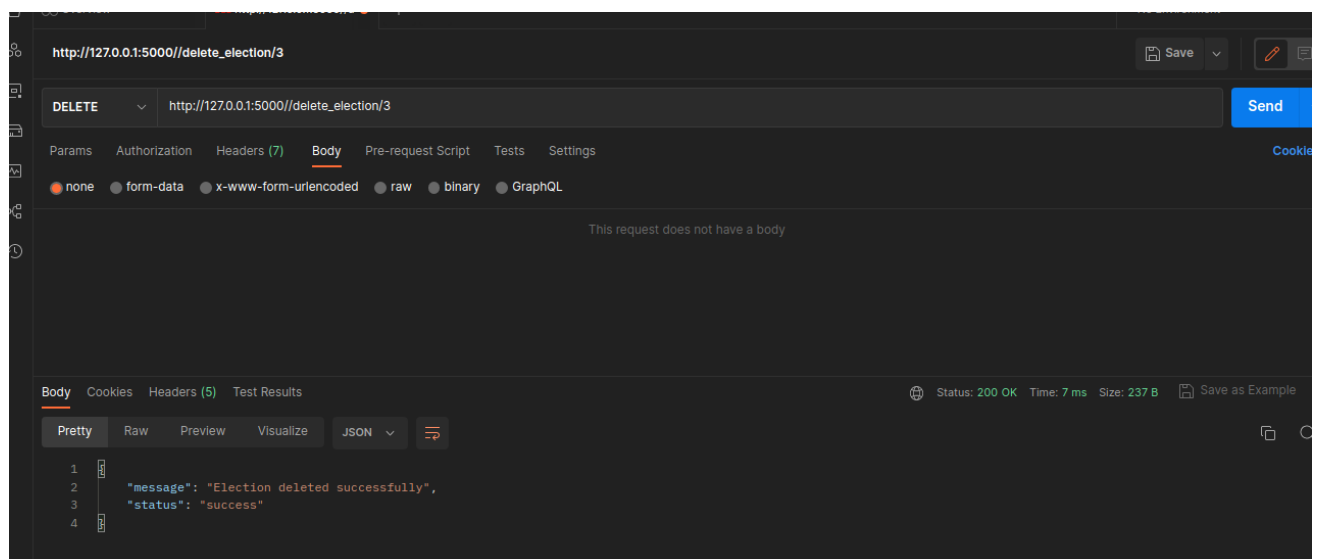
      RESPONSE:
a. **The resource was found and deleted successfully**
      RESPONSE:
      200(ok)

```
{
    "message": "Election deleted successfully",
    "status": "success"
}
```
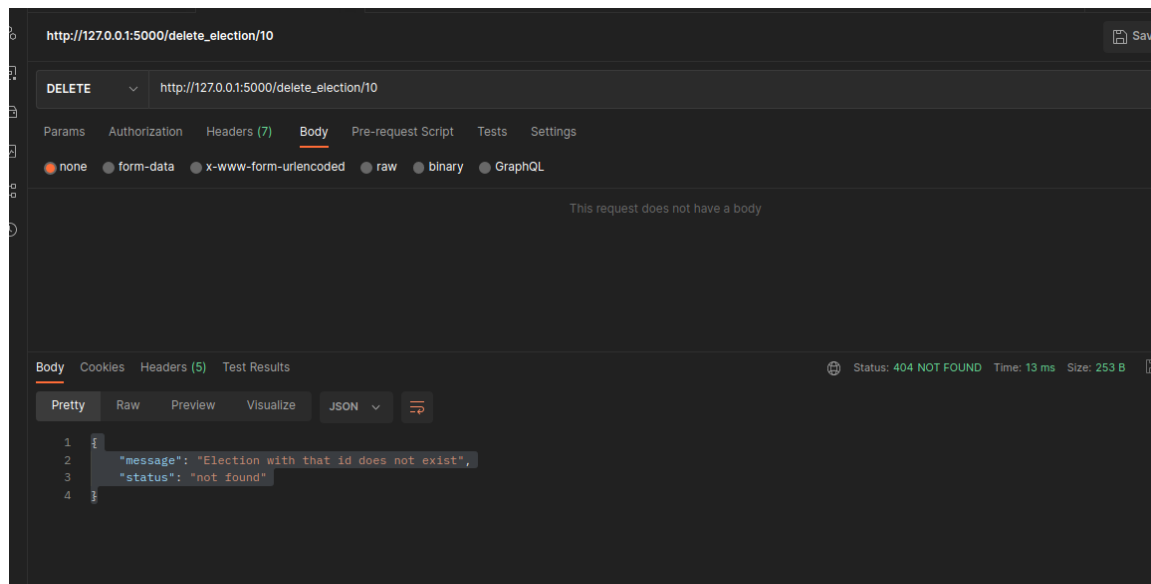
RESPONSE:
b. **The resource was not found during deletion**
   RESPONSE:
   404(not found)

```
{
    "message": "Election with that id does not exist",
    "status": "not found"
}
```

7. RETRIEVING AN ELECTION BY ELECTION ID

REQUEST
POST/`/get_election/<int:id>`

RESPONSE
200(ok)

```
{
    "data": [
        true,
        {
            "created_at": "2022-03-27T12:00:00Z",
            "description": "Election to choose the next student council leaders.",
            "election_id": 1,
            "election_type": "General",
            "end_time": "2022-04-01T18:00:00Z",
            "name": "Student Council Elections",
            "start_time": "2022-04-01T08:00:00Z",
            "status": "Scheduled",
            "updated_at": "2022-03-27T12:00:00Z"
        }
    ],
    "message": "successfull",
    "status": "success"
}
```
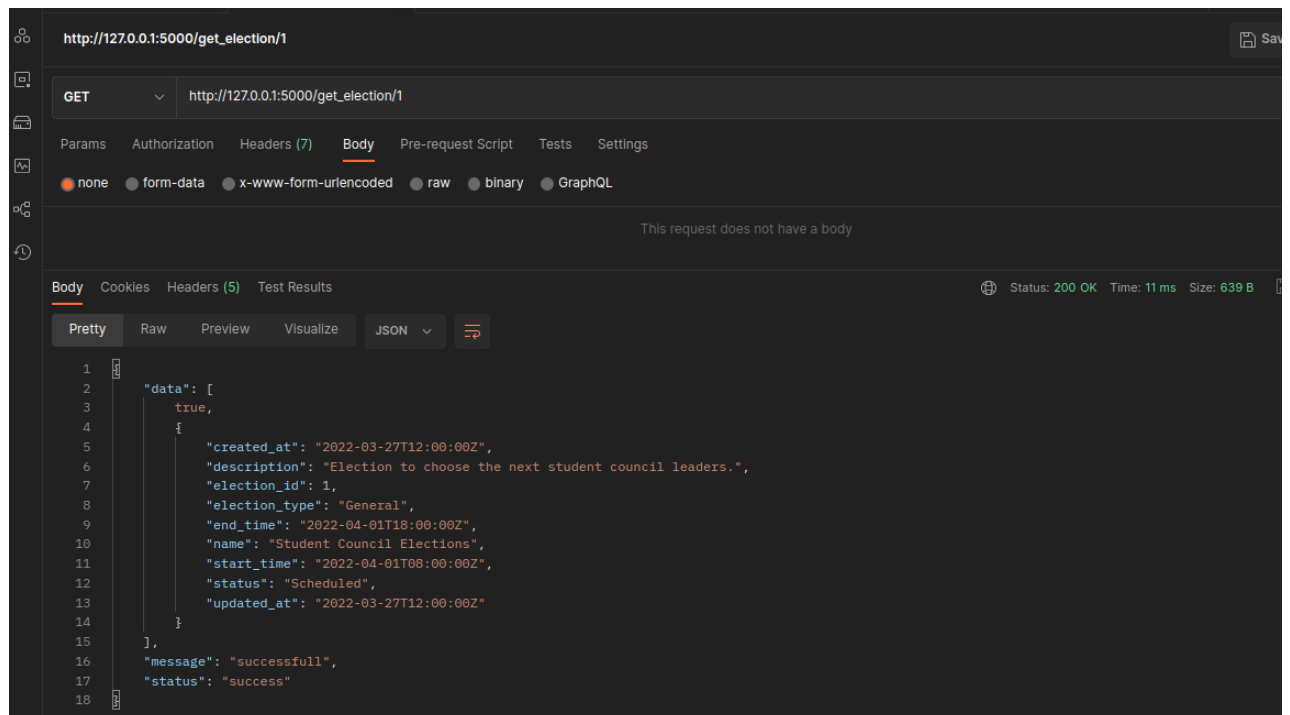
8. CASTING A VOTE

REQUEST:
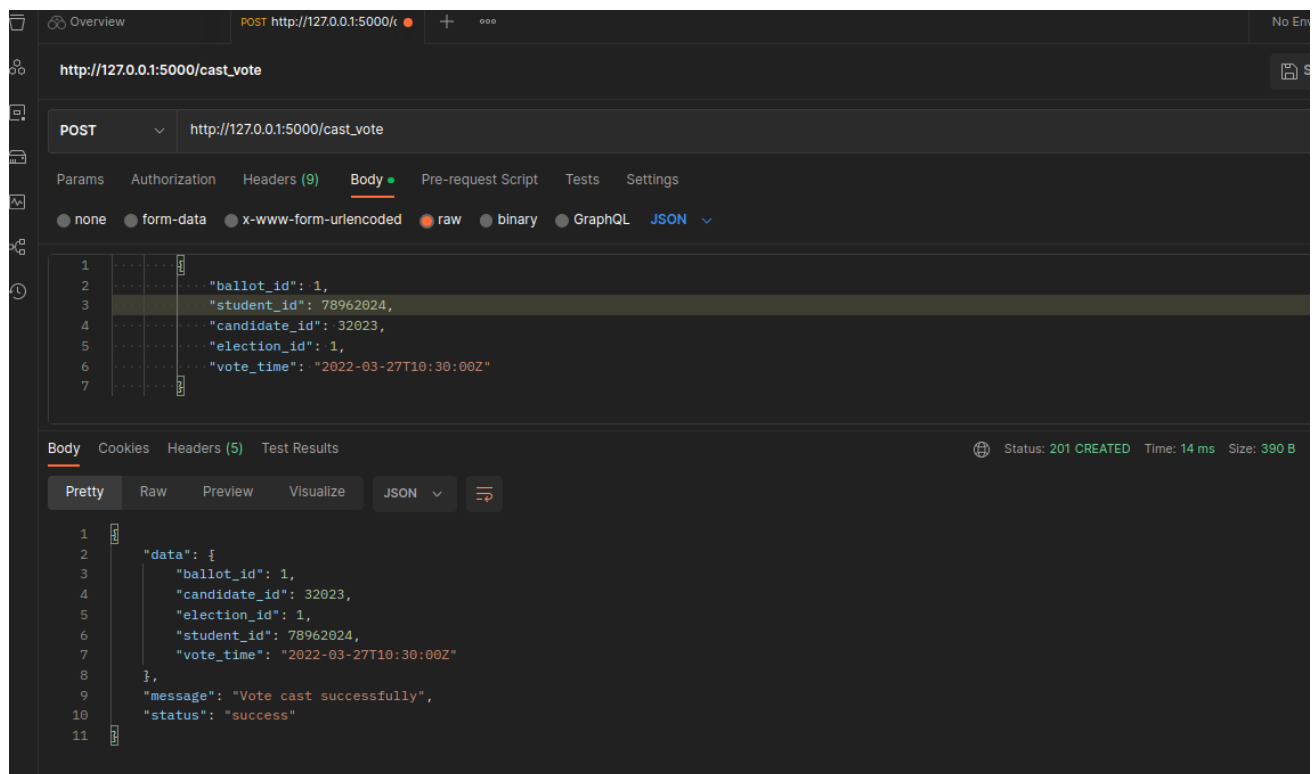**POST**`/cast_vote`

REQUEST BODY

```
{
    "ballot_id": 1,
    "student_id": 78962024,
    "candidate_id": 32023,
    "election_id": 1,
    "vote_time": "2022-03-27T10:30:00Z"
}
```

RESPONSE
If voter exists, candidate exists, election exist and voter has not already voted in that particular election:
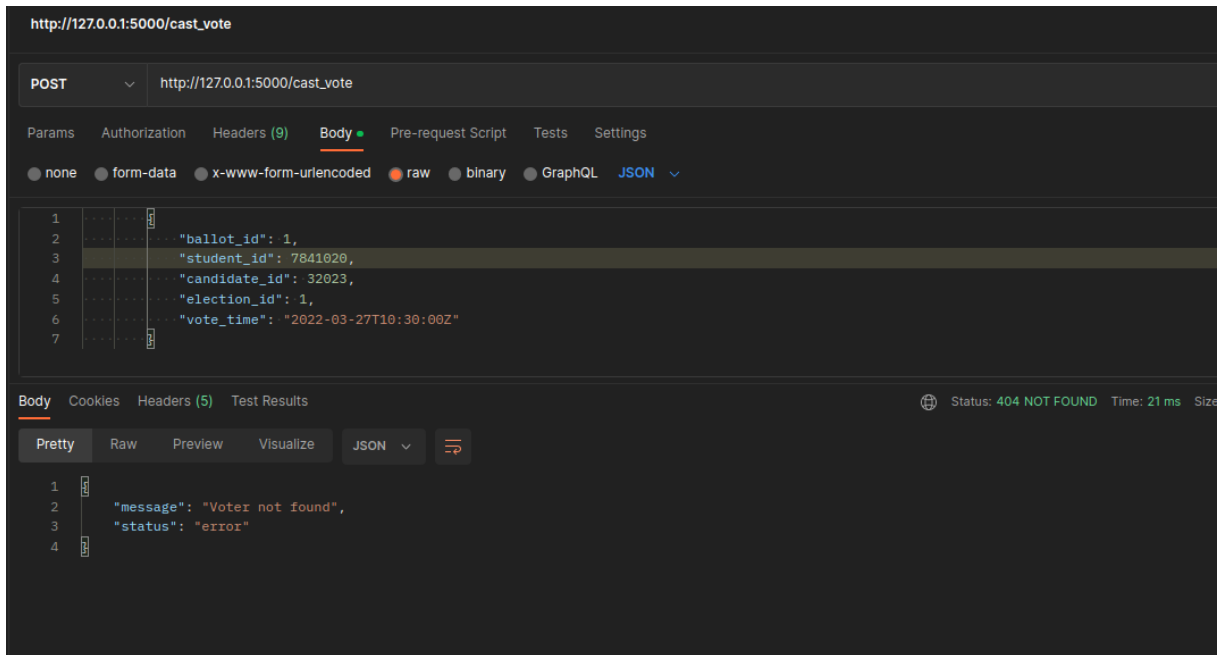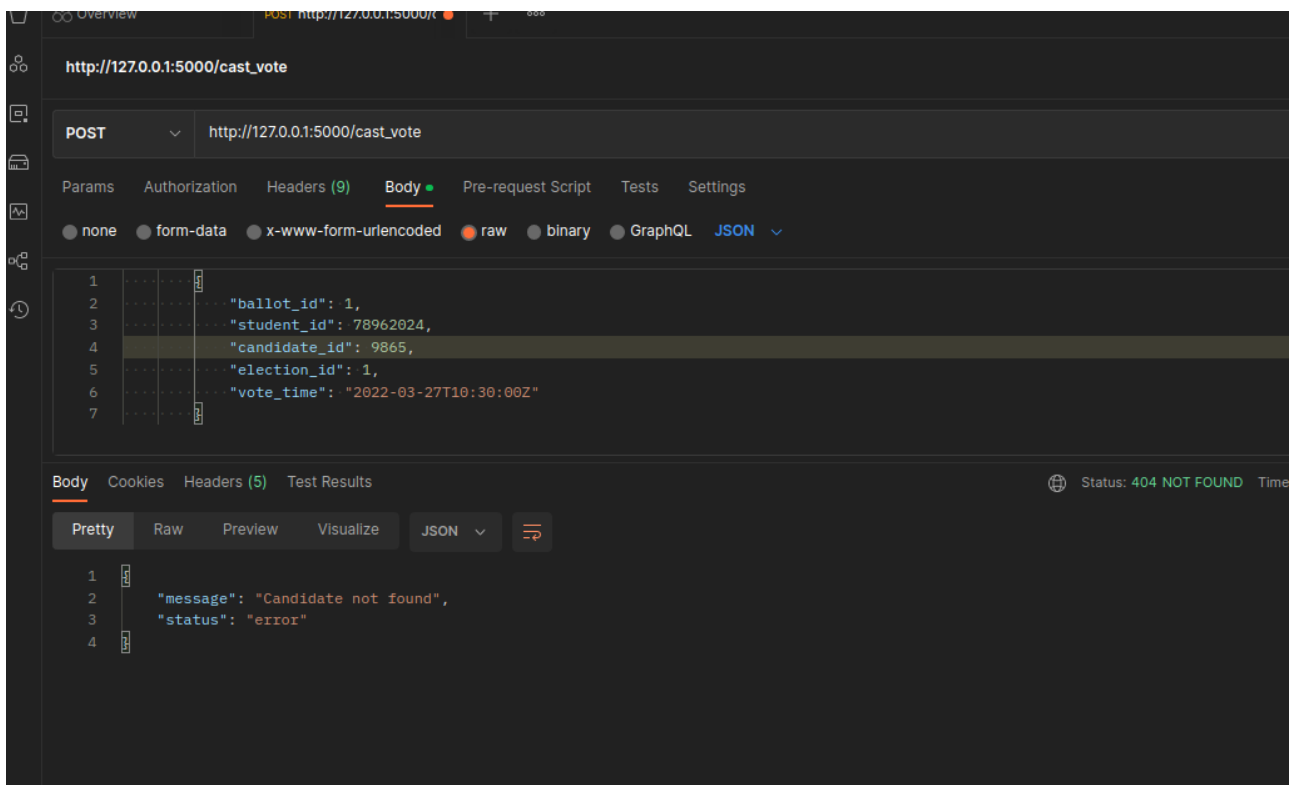Status code: **201(created)**

If voter does not exist:

404(Not found)

```
{
    "message": "Voter not found",
    "status": "error"
}
```

If candidate does not exist:
404(Not found)

If no election is found with the provided election id:

404 (Not found)

```
{
    "message": "Election not found",
    "status": "error"
}
```