

# SERVICE LAYER DESIGN

## SERVICE ENDPOINTS

### POSTS

Route	HTTP Verb	Purpose	Parameters
/posts	GET	Fetches all posts	
/posts/[id]	GET	Fetches a single post	
/posts	POST	Publishes a post	title, body
/posts/[id]	PATCH	Updates a post	title, body
/posts/[id]	DELETE	Deletes a post	

### USERS

Route	HTTP Verb	Purpose	Parameters
/users	GET	Fetches all users	
/users/[id]	GET	Fetches a single user	
/users	POST	Adds a user	username, password, email
/users/[id]	PATCH	Updates a user	username, password, email
/users/[id]	DELETE	Deletes a user	

### ERROR CODES

- 200: OK
- 201: Created
- 400: Bad Request
- 401: Unauthorized
- 403: Forbidden
- 404: Not Found

## EXAMPLE REQUESTS

### POSTS

```
FETCH ALL POSTS
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:3000/posts", requestOptions)
  .then(response => response.text())
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

```
.then(result => console.log(result))
.catch(error => console.log('error', error));
```

---

## RESPONSE

Status: 200 OK

```
[
  {
    "image": "",
    "_id": "5f050341bd906f3e50b7aa95",
    "title": "Post Title",
    "body": "Post Content",
    "timeStamp": "2020-07-07T17:10:20.546Z",
    "__v": 0
  },
  {
    "_id": "5f051379ce1bf360e8338ed7",
    "title": "Fix that issue now before your computer blows up",
    "body": "A text blurb",
    "timeStamp": "2020-07-08T00:29:45.400Z",
    "__v": 0
  },
  {
    "_id": "5f0514eb53c479a47c7750d1",
    "title": "That Puter is broken",
    "body": "A text blurb",
    "timeStamp": "2020-07-08T00:35:55.894Z",
    "__v": 0
  }
]
```

---

## FETCH ALL POSTS – ERROR

```
var requestOptions = {
  method: "GET",
  redirect: "follow",
};

fetch("http://localhost:3000/post", requestOptions)
  .then((response) => response.text())
  .then((result) => console.log(result))
  .catch((error) => console.log("error", error));
```

---

## RESPONSE

Status: 404 Not Found

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

```
<!DOCTYPE html>
<html Lang="en">
  <head>
    <meta charset="utf-8">
    <title>Error</title>
  </head>
  <body>
    <pre>Cannot GET /post</pre>
  </body>
</html>
```

---

## FETCH SINGLE POST

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:3000/posts/5f04ac7c082bf89d94b91329", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

---

## RESPONSE

Status: 200 OK

```
{
  "_id": "5f04ac7c082bf89d94b91329",
  "title": "Test",
  "body": "Test",
  "timeStamp": "2020-07-07T17:10:20.546Z",
  "__v": 0
}
```

---

## PUBLISH A POST

```
var raw = "{
  \n  \"title\": \"That Puter is broken\",
  \n  \"body\": \"A text blurb\"
  \n}";

var requestOptions = {
  method: 'POST',
  body: raw,
  redirect: 'follow'
};

fetch("http://localhost:3000/posts", requestOptions)
  .then(response => response.text())
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

```
.then(result => console.log(result))
.catch(error => console.log('error', error));
```

---

## RESPONSE

Status: 200 OK

```
{
  "_id": "5f051d64b31be9833c1cc406",
  "title": "That Puter is broken",
  "body": "A text blurb",
  "timeStamp": "2020-07-08T01:12:04.151Z",
  "__v": 0
}
```

---

## UPDATE A POST

```
var raw = "{
  \"title\": \"Fix that issue now before your computer blows up\",
  \"body\": \"A text blurb\"
}";

var requestOptions = {
  method: 'PATCH',
  body: raw,
  redirect: 'follow'
};

fetch("http://localhost:3000/posts/5f051379ce1bf360e8338ed7", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

---

## RESPONSE

Status: 200 OK

```
{
  "n": 1,
  "nModified": 1,
  "opTime": {
    "ts": "6846900918978347009",
    "t": 4
  },
  "electionId": "7ffffffff00000000000000004",
  "ok": 1,
  "$clusterTime": {
    "clusterTime": "6846900918978347009",
    "signature": {
      "hash": "MSVn0EW2nWtF5dKgTncUnM1HnA4=",
      "keyId": "6845936282208567299"
    }
  }
}
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

```
    }  
  },  
  "operationTime": "6846900918978347009"  
}
```

---

## UPDATE A POST ERROR

```
var raw = "{  
  \n    \"titl: \"Fix that issue now before your computer blows up\",  
  \n    \"bod\": \"A text blurp\"  
  \n}";  
  
var requestOptions = {  
  method: 'PATCH',  
  body: raw,  
  redirect: 'follow'  
};  
  
fetch("http://localhost:3000/posts/5f051379ce1bf360e8338ed7", requestOptions)  
  .then(response => response.text())  
  .then(result => console.log(result))  
  .catch(error => console.log('error', error));
```

---

## RESPONSE

Status: 400 Bad Request

```
<!DOCTYPE html>  
<html Lang="en">  
  <head>  
    <meta charset="utf-8">  
    <title>Error</title>  
  </head>  
  <body>  
    <pre>SyntaxError: Unexpected token F in JSON at position 16  
  
      <br> &nbsp; &nbsp; &nbsp;at JSON.parse (&lt;anonymous&gt;)  
  
      <br> &nbsp; &nbsp; &nbsp;at parse (C:\Users\muzz\OneDrive\Documents\MuzzTechBytes\node_modules  
body-parser\lib\types\json.js:89:19)  
  
      <br> &nbsp; &nbsp; &nbsp;at C:\Users\muzz\OneDrive\Documents\MuzzTechBytes\node_modules\body-  
parser\lib\read.js:121:18  
  
      <br> &nbsp; &nbsp; &nbsp;at invokeCallback (C:\Users\muzz\OneDrive\Documents\MuzzTechBytes\nod  
e_modules\raw-body\index.js:224:16)  
  
      <br> &nbsp; &nbsp; &nbsp;at done (C:\Users\muzz\OneDrive\Documents\MuzzTechBytes\node_modules\  
raw-body\index.js:213:7)
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

```
<br> &nbsp; &nbsp; &nbsp;at IncomingMessage.onEnd (C:\Users\muzz\OneDrive\Documents\MuzzTechBytes\node_modules\raw-body\index.js:273:7)

<br> &nbsp; &nbsp; &nbsp;at IncomingMessage.emit (events.js:323:22)

<br> &nbsp; &nbsp; &nbsp;at endReadableNT (_stream_readable.js:1204:12)

<br> &nbsp; &nbsp; &nbsp;at processTicksAndRejections (internal/process/task_queues.js:84:21)

</pre>
</body>
</html>
```

---

## DELETE A POST

```
var requestOptions = {
  method: "DELETE",
  redirect: "follow",
};

fetch("http://localhost:3000/posts/5f051379ce1bf360e8338ed7", requestOptions)
  .then((response) => response.text())
  .then((result) => console.log(result))
  .catch((error) => console.log("error", error));
```

---

## RESPONSE

Status: 200 Ok

```
{
  "n": 1,
  "opTime": {
    "ts": "6846914667168661505",
    "t": 4
  },
  "electionId": "7ffffffff00000000000000004",
  "ok": 1,
  "$clusterTime": {
    "clusterTime": "6846914667168661505",
    "signature": {
      "hash": "Ufy01G/czo2YS84x+TPJe7ZmfRA=",
      "keyId": "6845936282208567299"
    }
  },
  "operationTime": "6846914667168661505",
  "deletedCount": 1
}
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

## USERS

### FETCH ALL USERS

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:3000/users", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

### RESPONSE

Status: 200 Ok

```
[
  {
    "_id": "5f052496eacba179ac89bcb6",
    "username": "Muzz826",
    "password": "willbeencrypted",
    "email": "muzz826@gmail.com",
    "__v": 0
  },
  {
    "_id": "5f052508eacba179ac89bcb7",
    "username": "Atlas0408",
    "password": "willbeencrypted",
    "email": "purr@meow.com",
    "__v": 0
  },
  {
    "_id": "5f052523eacba179ac89bcb8",
    "username": "Gizmo",
    "password": "willbeencrypted",
    "email": "grr@bark.com",
    "__v": 0
  }
]
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

---

## ADD A USER

```
var raw = "{
  \n  \"username\" : \"Muzz826\",
  \n  \"password\" : \"willbeencrypted\",
  \n  \"email\" : \"muzz826@gmail.com\"
\n}";

var requestOptions = {
  method: 'POST',
  body: raw,
  redirect: 'follow'
};

fetch("http://localhost:3000/users", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

---

## RESPONSE

Status: 200 OK

```
{
  "_id": "5f052496eacba179ac89bcb6",
  "username": "Muzz826",
  "password": "willbeencrypted",
  "email": "muzz826@gmail.com",
  "__v": 0
}
```

---

## FETCH A SINGLE USER

```
var requestOptions = {
  method: 'GET',
  redirect: 'follow'
};

fetch("http://localhost:3000/users/5f052496eacba179ac89bcb6", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```



Name: Ben Muzzy

Assignment: Design: Service Layer(s)

---

## RESPONSE

Status 200 Ok

```
{
  "_id": "5f052496eacba179ac89bcb6",
  "username": "Muzz826",
  "password": "willbeencrypted",
  "email": "muzz826@gmail.com",
  "__v": 0
}
```

---

## UPDATE A USER

```
var raw = "{
  \n    \"username\": \"Gizmo818\"
  \n}";

var requestOptions = {
  method: 'PATCH',
  body: raw,
  redirect: 'follow'
};

fetch("http://localhost:3000/users/5f052523eacba179ac89bcb8", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

---

## RESPONSE

Status 200 Ok

```
"n": 1,
"nModified": 1,
"opTime": {
  "ts": "6846921045195096065",
  "t": 4
},
"electionId": "7fffffff0000000000000004",
"ok": 1,
"$clusterTime": {
  "clusterTime": "6846921045195096065",
  "signature": {
    "hash": "M8QBrDrG4NX/OtD80gKKrsrEqYs=",
    "keyId": "6845936282208567299"
  }
},
```

Name: Ben Muzzy

Assignment: Design: Service Layer(s)

```
"operationTime": "6846921045195096065"
}
```

---

## DELETE A USER

```
var requestOptions = {
  method: 'DELETE',
  redirect: 'follow'
};

fetch("http://localhost:3000/users/5f052547eacba179ac89bcb9", requestOptions)
  .then(response => response.text())
  .then(result => console.log(result))
  .catch(error => console.log('error', error));
```

---

## RESPONSE

Status 200 Ok

```
{
  "n": 1,
  "opTime": {
    "ts": "6846921470396858369",
    "t": 4
  },
  "electionId": "7fffffffff0000000000000004",
  "ok": 1,
  "$clusterTime": {
    "clusterTime": "6846921470396858369",
    "signature": {
      "hash": "TCojsau3vG3K/2PeozYUUrUgUoug=",
      "keyId": "6845936282208567299"
    }
  },
  "operationTime": "6846921470396858369",
  "deletedCount": 1
}
```

DIAGRAMS

User Interface Requests Flow

