

## API Content from the Document Created By Wilson, with reduced set of API's

Please note that there would be some changes on going (minor) as we are implementing the different features.

If you have any comments regarding the API's please mail [manoj.dhoble@gmail.com](mailto:manoj.dhoble@gmail.com)

\*\*\*\*\*

### Response for Add Requests and Dealing with ObjectId:

if Success:

```
output = {"success" : True, "id": objectIdFromMongoAsString}
```

else:

```
output = {"success" : False}
```

When an item is inserted in mongo, its objectId is created. But mongo stores it as an object, which cannot be serialized to JSON. Hence, on each insert, convert the generated ObjectId to string, and send it in the response. (The mongo insert function returns the generated ObjectId)

e.g.

```
obj_id = self.course.insert(json)
```

```
obj_id = str(obj_id)
```

Now, this obj\_id is a string and hence JSON Serializable.

Similarly, when you get/update an item, you need to convert the id string that is passed in the request url to its ObjectId before performing the db operation.

e.g

```
obj_id = ObjectId(id)
```

```
query = {"_id": obj_id}
```

Now u can perform the db operation according to this "query"

Also, when you perform the "Get" request, you should remove the "\_id" from the entity returned, or simply

```
for c in self.quiz.find(query):
```

```
    return {"id": idFromRequestUrl, "courseId": c["courseId"], "questions": c["questions"]}
```

This should simplify things for everyone :)

\*\*\*\*\*

For Login/ Logout/ User Authentication, we are following Django's Inbuild User Authentication Mechanism : <https://docs.djangoproject.com/en/1.0/topics/auth/>

## Structure of API

### User

#### Create User

Request

Response

#### Get User

Usage

Request

Response

#### Update User

Usage

Request

Response

#### Delete User

Request

Response

#### Enroll Course

Request

Response

#### Drop Course

Request

Response

### Category

#### Add Category

Request

Response

#### Get Category

Request

Response

#### List Category

Request

Response

### Course

#### Add Course

Request

Response

#### Get Course

Request

Response

#### List Course

Request

[Response](#)

[Update Course](#)

[Request](#)

[Response](#)

[Delete Course](#)

[Request](#)

[Response](#)

[Quiz](#)

[Add Quiz](#)

[Request](#)

[Response](#)

[Get Quiz](#)

[Request](#)

[Response](#)

[List Quiz](#)

[Request](#)

[Response](#)

[Update Quiz](#)

[Request](#)

[Response](#)

[Delete Quiz](#)

[Request](#)

[Response](#)

[Announcement](#)

[Add Announcement](#)

[Request](#)

[Response](#)

[List Announcement](#)

[Request](#)

[Response](#)

[Get Announcement](#)

[Request](#)

[Response](#)

[Update Announcement](#)

[Request](#)

[Response](#)

[Delete Announcement](#)

[Request](#)

[Response](#)

[Discussion](#)

[Add Discussion](#)

[Request](#)

[Response](#)

[List Discussion](#)

[Request](#)  
[Response](#)  
[Get Discussion](#)  
[Request](#)  
[Response](#)  
[Update Discussion](#)  
[Request](#)  
[Response](#)  
[Delete Discussion](#)  
[Request](#)  
[Response](#)  
[Message](#)  
[Add Message](#)  
[Request](#)  
[Response](#)  
[List Message](#)  
[Request](#)  
[Response](#)  
[Update Message](#)  
[Request](#)  
[Response](#)  
[Delete Message](#)  
[Request](#)  
[Response](#)  
[Database Collections](#)  
[usercollection](#)  
[catagorycollection](#)  
[coursecollection](#)  
[discussioncollection](#)

## Structure of API

/user

- 1) create
- 2) sign In
- 3) sign Out
- 4) update
- 5) delete
- 6) get
- 7) enroll
- 8) drop

/category

- 1) add
- 2) list
- 3) get

/course

- 1) add
- 2) edit
- 3) list
- 4) delete

/course/quiz

- 1) add --> only the person adding course can add quiz.
- 2) edit --> only the person adding course can add quiz.
- 3) list
- 4) get
- 5) delete --> only the person adding course can add quiz.

/course/announcement

- 1) add --> only the person adding course can add announcement.
- 2) edit --> only the person adding course can add announcement.
- 3) list
- 4) delete --> only the person adding course can add announcement.
- 5) get (do we need it ?)

/course/discussion

- 1) add
- 2) edit
- 3) list
- 4) delete
- 5) get

/course/discussion/message

- 1) add
- 2) edit
- 3) list
- 4) delete

## User

### Create User

The password should be always hashed by SHA-1, lower-cased hex string.

#### Request

| <i>Method</i> | <i>URL</i>            | <i>Request Body</i> |
|---------------|-----------------------|---------------------|
| POST          | http://host:port/user | Register User JSON  |

#### Response

| <i>Status Code</i> | <i>Response Body</i> | <i>Condition</i> |
|--------------------|----------------------|------------------|
|--------------------|----------------------|------------------|

|                |              |                     |
|----------------|--------------|---------------------|
| 201 (Created)  | Success JSON | Success             |
| 409 (Conflict) | Error JSON   | Email is duplicated |

## Get User

### Usage

It used for any situation when servers or clients need the user's information by given email.

### Request

| <i>Method</i> | <i>URL</i>                           | <i>Request Body</i> |
|---------------|--------------------------------------|---------------------|
| GET           | http://host:port/user/: <b>email</b> | -                   |

### Response

| <i>Status Code</i>          | <i>Response Body</i>          | <i>Condition</i> |
|-----------------------------|-------------------------------|------------------|
| 200 (OK)                    | User JSON <b>w/o password</b> | Success          |
| 400 (Bad Request)           | Error JSON                    | Email is invalid |
| 404 (Not Found)             | Error JSON                    | User not found   |
| 500 (Internal Server Error) | Error JSON                    | Other Errors     |

## Update User

### Usage

Can be used for both creating and updating user, since the email address is unique.

### Request

| <i>Method</i> | <i>URL</i>   | <i>Request Body</i> |
|---------------|--|---------------------|
| PUT           | http://host:port/user/ <b>update</b> /: <b>email</b> | User JSON *         |

\*Password is required for creating new user, but it will be ignored for updating.

### Response

| <i>Status Code</i>          | <i>Response Body</i>          | <i>Condition</i>              |
|-----------------------------|-------------------------------|-------------------------------|
| 201 (Created)               | User JSON w/o password        | Create new user if not exists |
| 200 (OK)                    | User JSON <b>w/o password</b> | Update existing user          |
| 400 (Bad Request)           | Error JSON                    | Email or JSON is invalid      |
| 500 (Internal Server Error) | Error JSON                    | Other Errors                  |

## Delete User

### Request

| <i>Method</i> | <i>URL</i>                           | <i>Request Body</i> |
|---------------|--------------------------------------|---------------------|
| DELETE        | http://host:port/user/: <b>email</b> | -                   |

### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Success JSON         | Delete success   |
| 400 (Bad Request)           | Error JSON           | Email is invalid |
| 404 (Not Found)             | Error JSON           | User not found   |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

## Enroll Course

### Request

| <i>Method</i> | <i>URL</i>                     | <i>Request Body</i>                              |
|---------------|--------------------------------|--|
| PUT           | http://host:port/course/enroll | email=: <b>email</b> &courseid=: <b>courseid</b> |

### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Success JSON         | Success          |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

## Drop Course

### Request

| <i>Method</i> | <i>URL</i>                   | <i>Request Body</i>                              |
|---------------|------------------------------|--|
| PUT           | http://host:port/course/drop | email=: <b>email</b> &courseid=: <b>courseid</b> |

### Response

| <i>Status Code</i> | <i>Response Body</i> | <i>Condition</i> |
|--------------------|----------------------|------------------|
| 200 (OK)           | Success JSON         | Success          |

|                             |            |              |
|-----------------------------|------------|--------------|
| 500 (Internal Server Error) | Error JSON | Other Errors |
|-----------------------------|------------|--------------|

## Category

### Add Category

#### Request

| <i>Method</i> | <i>URL</i>                | <i>Request Body</i>         |
|---------------|---------------------------|-----------------------------|
| POST          | http://host:port/category | Category JSON <b>w/o ID</b> |

#### Response

| <i>Status Code</i>          | <i>Response Body</i>       | <i>Condition</i>   |
|-----------------------------|----------------------------|--------------------|
| 201 (Created)               | Category JSON <b>w/ ID</b> | Success            |
| 409 (Conflict)              | Error JSON                 | Name is duplicated |
| 500 (Internal Server Error) | Error JSON                 | Other Errors       |

### Get Category

#### Request

| <i>Method</i> | <i>URL</i>                    | <i>Request Body</i> |
|---------------|-------------------------------|---------------------|
| GET           | http://host:port/category/:id | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>   |
|-----------------------------|----------------------|--------------------|
| 200 (OK)                    | Category JSON        | Success            |
| 400 (Bad Request)           | Error JSON           | ID is invalid      |
| 404 (Not Found)             | Error JSON           | Category not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors       |

### List Category

#### Request

| <i>Method</i> | <i>URL</i>                     | <i>Request Body</i> |
|---------------|--------------------------------|---------------------|
| GET           | http://host:port/category/list | -                   |

#### Response



| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Category Array JSON  | Success          |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

## Course

### Add Course

#### Request

| <i>Method</i> | <i>URL</i>              | <i>Request Body</i>       |
|---------------|-------------------------|---------------------------|
| POST          | http://host:port/course | Course JSON <b>w/o ID</b> |

#### Response

| <i>Status Code</i>          | <i>Response Body</i>     | <i>Condition</i> |
|-----------------------------|--------------------------|------------------|
| 201 (Created)               | Course JSON <b>w/ ID</b> | Success          |
| 500 (Internal Server Error) | Error JSON               | Other Errors     |

### Get Course

#### Request

| <i>Method</i> | <i>URL</i>                  | <i>Request Body</i> |
|---------------|-----------------------------|---------------------|
| GET           | http://host:port/course/:id | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Course JSON          | Success          |
| 400 (Bad Request)           | Error JSON           | ID is invalid    |
| 404 (Not Found)             | Error JSON           | Course not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

### List Course

#### Request

| <i>Method</i> | <i>URL</i> | <i>Request Body</i> |
|---------------|------------|---------------------|
|---------------|------------|---------------------|

|     |                              |   |
|-----|------------------------------|---|
| GET | http://host:port/course/list | - |
|-----|------------------------------|---|

#### Response

It should redirect to the URL of the course list page using 301 (Moved Permanently) with URL in Location Header. If there is any error, display the error page instead.

## Update Course

#### Request

| <i>Method</i> | <i>URL</i>                                 | <i>Request Body</i> |
|---------------|--|---------------------|
| PUT           | http://host:port/course/ <b>update/:id</b> | Course JSON         |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>      |
|-----------------------------|----------------------|-----------------------|
| 200 (OK)                    | Course JSON          | Update success        |
| 400 (Bad Request)           | Error JSON           | ID or JSON is invalid |
| 404 (Not Found)             | Error JSON           | Course not found      |
| 500 (Internal Server Error) | Error JSON           | Other Errors          |

## Delete Course

#### Request

| <i>Method</i> | <i>URL</i>                          | <i>Request Body</i> |
|---------------|-------------------------------------|---------------------|
| DELETE        | http://host:port/course/: <b>id</b> | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Success JSON         | Delete success   |
| 400 (Bad Request)           | Error JSON           | ID is invalid    |
| 404 (Not Found)             | Error JSON           | User not found   |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

## Quiz

### Add Quiz

### Request

| Method | URL                      | Request Body            |
|--------|--------------------------|-------------------------|
| POST   | http://host:port/quizzes | Quiz JSON <b>w/o ID</b> |

### Response

| Status Code                 | Response Body          | Condition    |
|-----------------------------|------------------------|--------------|
| 201 (Created)               | Quiz JSON <b>w/ ID</b> | Success      |
| 500 (Internal Server Error) | Error JSON             | Other Errors |

## Get Quiz

### Request

| Method | URL                       | Request Body |
|--------|---------------------------|--------------|
| GET    | http://host:port/quiz/:id | -            |

### Response

| Status Code                 | Response Body | Condition      |
|-----------------------------|---------------|----------------|
| 200 (OK)                    | Quiz JSON     | Success        |
| 400 (Bad Request)           | Error JSON    | ID is invalid  |
| 404 (Not Found)             | Error JSON    | Quiz not found |
| 500 (Internal Server Error) | Error JSON    | Other Errors   |

## List Quiz

### Request

| Method | URL                    | Request Body |
|--------|------------------------|--------------|
| GET    | http://host:port/quiz/ | -            |

### Response

| Status Code                 | Response Body | Condition    |
|-----------------------------|---------------|--------------|
| 200 (OK)                    | Quiz JSON     | Success      |
| 500 (Internal Server Error) | Error JSON    | Other Errors |

## Update Quiz

### Request

| <i>Method</i> | <i>URL</i>  | <i>Request Body</i> |
|---------------|---|---------------------|
| PUT           | http://host:port/quiz/ <b>update/</b><br><b>:id</b> | Quiz JSON           |

### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>      |
|-----------------------------|----------------------|-----------------------|
| 200 (OK)                    | Quiz JSON            | Update success        |
| 400 (Bad Request)           | Error JSON           | ID or JSON is invalid |
| 404 (Not Found)             | Error JSON           | Quiz not found        |
| 500 (Internal Server Error) | Error JSON           | Other Errors          |

## Delete Quiz

### Request

| <i>Method</i> | <i>URL</i>                        | <i>Request Body</i> |
|---------------|-----------------------------------|---------------------|
| DELETE        | http://host:port/quiz/ <b>:id</b> | -                   |

### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Success JSON         | Delete success   |
| 400 (Bad Request)           | Error JSON           | ID is invalid    |
| 404 (Not Found)             | Error JSON           | User not found   |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

## Announcement

### Add Announcement

### Request

| <i>Method</i> | <i>URL</i>                         | <i>Request Body</i>             |
|---------------|------------------------------------|---------------------------------|
| POST          | http://host:port/<br>announcements | Announcement JSON <b>w/o ID</b> |

#### Response

| <i>Status Code</i>          | <i>Response Body</i>           | <i>Condition</i> |
|-----------------------------|--------------------------------|------------------|
| 201 (Created)               | Announcement JSON <b>w/ ID</b> | Success          |
| 500 (Internal Server Error) | Error JSON                     | Other Errors     |

### List Announcement

#### Request

| <i>Method</i> | <i>URL</i>                             | <i>Request Body</i> |
|---------------|--|---------------------|
| GET           | http://host:port/<br>announcement/list | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Announcement JSON    | Success          |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

### Get Announcement

#### Request

| <i>Method</i> | <i>URL</i>                                    | <i>Request Body</i> |
|---------------|---|---------------------|
| GET           | http://host:port/<br>announcement/: <b>id</b> | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>       |
|-----------------------------|----------------------|------------------------|
| 200 (OK)                    | Announcement JSON    | Success                |
| 400 (Bad Request)           | Error JSON           | ID is invalid          |
| 404 (Not Found)             | Error JSON           | Announcement not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors           |

### Update Announcement

#### Request

| <i>Method</i> | <i>URL</i>   | <i>Request Body</i> |
|---------------|--|---------------------|
| PUT           | http://host:port/<br>announcement/ <b>update/:id</b> | Announcement JSON   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>       |
|-----------------------------|----------------------|------------------------|
| 200 (OK)                    | Announcement JSON    | Update success         |
| 400 (Bad Request)           | Error JSON           | ID or JSON is invalid  |
| 404 (Not Found)             | Error JSON           | Announcement not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors           |

## Delete Announcement

#### Request

| <i>Method</i> | <i>URL</i>                                    | <i>Request Body</i> |
|---------------|---|---------------------|
| DELETE        | http://host:port/<br>announcement/ <b>:id</b> | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Success JSON         | Delete success   |
| 400 (Bad Request)           | Error JSON           | ID is invalid    |
| 404 (Not Found)             | Error JSON           | User not found   |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

## Discussion

### Get Discussion by ID

#### Request

| <i>Method</i> | <i>URL</i>                              | <i>Request Body</i> |
|---------------|---|---------------------|
| GET           | http://host:port/discussion/ <b>:id</b> | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>     |
|-----------------------------|----------------------|----------------------|
| 200 (OK)                    | Discussion JSON      | Success              |
| 400 (Bad Request)           | Error JSON           | ID is invalid        |
| 404 (Not Found)             | Error JSON           | Discussion not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors         |

### Get Discussion by Course ID

#### Request

| <i>Method</i> | <i>URL</i>   | <i>Request Body</i> |
|---------------|--|---------------------|
| GET           | http://host:port/course/<br>:course_id/discussion<br>http://host:port/discussion/<br>course/:course_id | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>     |
|-----------------------------|----------------------|----------------------|
| 200 (OK)                    | Discussion JSON      | Success              |
| 404 (Not Found)             | Error JSON           | Discussion not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors         |

### Search Discussions (Optional)

#### Request

| <i>Method</i> | <i>URL</i>  | <i>Request Body</i> |
|---------------|---|---------------------|
| GET           | http://host:port/discussions?<br>param=value&...<br>http://host:port/course/<br>:course_id/discussion | -                   |

#### Parameters

All parameters are optional, usually are specified by default.

|           |   |
|-----------|---|
| q         | A part of title, description, course name (like google search)              |
| course_id | Exactly the course id (to support discussion without course_id, use "none") |

|                 |  |
|-----------------|--|
| title           | Exactly the title  |
| created_by      | Exactly the creator's email  |
| created_from/to | The time span for created_at property  |
| updated_from/to | The time span for updated_at property, can be used for Sync  |
| offset          | For paging, the offset index of entity for this return, default 0  |
| limit           | For paging, the numbers of entity for this return, default 20  |
| order_by        | For ordering and paging (use fieldname.asc/desc, asc by default, e.g. updated_by.desc), default is order by id |

#### Response

| <i>Status Code</i>          | <i>Response Body</i>  | <i>Condition</i> |
|-----------------------------|-----------------------|------------------|
| 200 (OK)                    | Discussion Array JSON | Success          |
| 500 (Internal Server Error) | Error JSON            | Other Errors     |

### Add Discussion

#### Request

| Method | URL                          | Request Body                  |
|--------|------------------------------|-------------------------------|
| POST   | http://host:port/discussions | Discussion JSON <b>w/o ID</b> |

#### Response

| <i>Status Code</i>          | <i>Response Body</i>         | <i>Condition</i> |
|-----------------------------|------------------------------|------------------|
| 201 (Created)               | Discussion JSON <b>w/ ID</b> | Success          |
| 500 (Internal Server Error) | Error JSON                   | Other Errors     |

### Update Discussion

#### Request

| <i>Method</i> | <i>URL</i>                                     | <i>Request Body</i> |
|---------------|--|---------------------|
| PUT           | http://host:port/discussion/ <b>update/:id</b> | Discussion JSON     |



### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>      |
|-----------------------------|----------------------|-----------------------|
| 200 (OK)                    | Discussion JSON      | Update success        |
| 400 (Bad Request)           | Error JSON           | ID or JSON is invalid |
| 404 (Not Found)             | Error JSON           | Discussion not found  |
| 500 (Internal Server Error) | Error JSON           | Other Errors          |

### **Delete Discussion (Optional)**

#### Request

| <i>Method</i> | <i>URL</i>                      | <i>Request Body</i> |
|---------------|---------------------------------|---------------------|
| DELETE        | http://host:port/discussion/:id | -                   |

### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i> |
|-----------------------------|----------------------|------------------|
| 200 (OK)                    | Success JSON         | Delete success   |
| 400 (Bad Request)           | Error JSON           | ID is invalid    |
| 404 (Not Found)             | Error JSON           | User not found   |
| 500 (Internal Server Error) | Error JSON           | Other Errors     |

### **Get Messages of a Discussion**

#### Usage

Get all messages of a discussion with paging and delta sync support.

#### Request

| <i>Method</i> | <i>URL</i>  |
|---------------|---|
| GET           | http://host:port/discussion/:discussion_id/<br>messages?params... |

#### Parameters

All parameters are optional, usually are specified by default. If no parameter is given, return all.

|                 |   |
|-----------------|---|
| created_from/to | The time span for created_at property                       |
| updated_from/to | The time span for updated_at property, can be used for Sync |

|          |  |
|----------|--|
| offset   | For paging, the offset index of entity for this return, default 0  |
| limit    | For paging, the numbers of entity for this return, default 20  |
| order_by | For ordering and paging (use fieldname.asc/desc, asc by default, e.g. updated_by.desc), default is order by id.asc or created_at.desc? |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>     |
|-----------------------------|----------------------|----------------------|
| 200 (OK)                    | Message Array JSON   | Success              |
| 400 (Bad Request)           | Error JSON           | ID is invalid        |
| 404 (Not Found)             | Error JSON           | Discussion not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors         |

### **Get Message from a Discussion by ID**

#### Request

| <i>Method</i> | <i>URL</i>   | <i>Request Body</i> |
|---------------|--|---------------------|
| GET           | http://host:port/discussion/<br>:discussion_id/message/<br>:message_id | -                   |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>     |
|-----------------------------|----------------------|----------------------|
| 200 (OK)                    | Message JSON         | Success              |
| 400 (Bad Request)           | Error JSON           | ID is invalid        |
| 404 (Not Found)             | Error JSON           | Discussion not found |
| 500 (Internal Server Error) | Error JSON           | Other Errors         |

### **Add Message to a Discussion**

#### Request

| Method | URL | Request Body |
|--------|-----|--------------|
|--------|-----|--------------|

|      |  |                            |
|------|--|----------------------------|
| POST | http://host:port/discussion/<br><b>/add//:discussion_id/</b><br>messages | Message JSON <b>w/o ID</b> |
|------|--|----------------------------|

#### Response

| <i>Status Code</i>          | <i>Response Body</i>      | <i>Condition</i> |
|-----------------------------|---------------------------|------------------|
| 201 (Created)               | Message JSON <b>w/ ID</b> | Success          |
| 500 (Internal Server Error) | Error JSON                | Other Errors     |

### Update Message of a Discussion (Optional)

#### Request

| <i>Method</i> | <i>URL</i>  | <i>Request Body</i> |
|---------------|---|---------------------|
| PUT           | http://host:port/message/<br>update/:message_id/<br>discussion/:discussion_id | Message JSON        |

#### Response

| <i>Status Code</i>          | <i>Response Body</i> | <i>Condition</i>      |
|-----------------------------|----------------------|-----------------------|
| 200 (OK)                    | Message JSON         | Update success        |
| 400 (Bad Request)           | Error JSON           | ID or JSON is invalid |
| 404 (Not Found)             | Error JSON           | Discussion not found  |
| 500 (Internal Server Error) | Error JSON           | Other Errors          |

### Delete Message from a Discussion (Optional)

#### Request

| <i>Method</i> | <i>URL</i>  | <i>Request Body</i> |
|---------------|---|---------------------|
| DELETE        | http://host:port/message/<br>: <b>message_id</b> /discussion/<br>: <b>discussion_id</b> | -                   |

#### Response

| <i>Status Code</i> | <i>Response Body</i> | <i>Condition</i> |
|--------------------|----------------------|------------------|
| 200 (OK)           | Success JSON         | Delete success   |
| 400 (Bad Request)  | Error JSON           | ID is invalid    |
| 404 (Not Found)    | Error JSON           | User not found   |

|                             |            |              |
|-----------------------------|------------|--------------|
| 500 (Internal Server Error) | Error JSON | Other Errors |
|-----------------------------|------------|--------------|

## Database Collections

### usercollection

```
{
  "email": "test@google.com",
  "own": [
    "course1",
    "course2"
  ],
  "enrolled": [
    "course3",
    "course4"
  ],
  "quizzes": [
    {
      "quiz": "id1",
      "grade": 5,
      "submitDate": "DATE"
    },
    {
      "quiz": "id1",
      "grade": 5,
      "submitDate": "DATE"
    }
  ]
}
```

### catagorycollection

```
{
  "name": "NAME",
  "description": "desc",
  "createDate": "DATE",
  "status": 0
}
```

### coursecollection

```
{
  "category": "categoryId",
  "title": "introduction to algebra",
}
```

```
"section": 2,
"dept": "eng",
"term": "Spring",
"year": 2013,
"instructor": [
  {
    "name": "russe! Doe",
    "email": "russe!_doe@univ.com"
  }
],
"days": [
  "Monday",
  "Wednesday",
  "Friday"
],
"hours": [
  "8:00AM",
  "9:15:AM"
],
"Description": "My course",
"attachment": "PATH",
"version": "1"
}
```

#### **announcementcollection**

```
{
  "courseId": "courseId",
  "title": "Title",
  "description": "desc",
  "postDate": "DATE",
  "status": 0
}
```

#### **quizcollection**

```
{
  "courseId": "courseId",
  "questions": [
    {
      "question": "Que1",
      "options": [
        "option1",
        "option2"
      ],
      "answer": "option1",
    }
  ]
}
```

```

        "point": 1
    },
    {
        "question": "Que2",
        "options": [
            "option1",
            "option2"
        ],
        "answer": "option1",
        "point": 1
    }
]
}

```

## Discussion

```

{
    course_id(?): // optional, for discussion without course
    title: // optional, the topic
    created_by: // user email
    created_at: "2013-04-18T08:56:20.583Z" // ISO format
    updated_at: // default is the same as created_at
}

```

## Message

It is separated from Discussion to avoid the size grow too big and concurrency issue. It depends on your team's decision whether it should be embedded or not, but doubtful for large discussions.

```

{
    title: // optional
    content: // text
    discussion_id:
    created_by: // user email
    created_at: "2013-04-18T08:56:20.583Z" // ISO format
    updated_at: // default is the same as created_at
}

```

To use `_id` on Django we need to use template tags as filters