



## API Overview

---

### Table of Contents

Introduction.....	2
Http Post Parameters.....	2
Common HTTP error code responses .....	3
HTTP Error Code 400 Bad Request.....	3
HTTP Error Code 401 Unauthorized .....	3
HTTP Error Code 404 Not Found.....	3
HTTP Error Code 405 Method Not Allowed .....	4
HTTP Error Code 415 Unsupported Media Type .....	4
HTTP Error Code 500 Internal Server Error.....	4
Process to create a recurring invoice for a new client: .....	5
Sample Java code:.....	6

## Introduction

- RESTful JSON over https.
- UTF-8 encoding.
- Date format: ISO8601 UTC, yyyy-MM-dd'T'HH:mm:ssz e.g. 2015-05-23T00:00:00UTC.
- Amounts must be in the format 0.00 e.g. 1000.00
- API user is created under the General Settings tab -> Staff Management -> Create User. Select API User checkbox. Api Key will automatically be generated. Enter a password and save.

## Http Post Parameters

Http Request	Description
apikey	Created in staff management in the General Settings tab.  This must be set in the <b>http header request</b> for each call.
password	Created in staff management in the General Settings tab.  This must be set in the <b>http header request</b> for each call.
Http content	Mime Type: application/json  The JSON message in UTF-8 text format.

## Common HTTP error code responses

### HTTP Error Code 400 Bad Request

```
{
  "id":null,
  "status":"NOK",
  "messages":[
    "Unexpected character " expecting comma to separate OBJECT entries"
  ],
  "additionalProperties":{
  }
}
```

Please ensure your request is valid according to API documentation

### HTTP Error Code 401 Unauthorized

```
{
  "id":null,
  "status":"NOK",
  "messages":[
    "Wrong username or password"
  ],
  "additionalProperties":{
  }
}
```

Please ensure you provide a valid username and password

### HTTP Error Code 404 Not Found

```
{
  "id":null,
  "status":"NOK",
  "messages":[
    "The requested resource is not available"
  ],
  "additionalProperties":{
  }
}
```

Please ensure your request URL is valid

### **HTTP Error Code 405 Method Not Allowed**

```
{
  "id":null,
  "status":"NOK",
  "messages":[
    "The specified HTTP method is not allowed for the requested
resource"
  ],
  "additionalProperties":{

}
}
```

Please ensure your request method is correct, eg. POST, GET, etc

### **HTTP Error Code 415 Unsupported Media Type**

```
{
  "id":null,
  "status":"NOK",
  "messages":[
    "The server refused this request because the request entity is in a
format not supported by the requested resource for the requested method"
  ],
  "additionalProperties":{

}
}
```

Please ensure you user HTTP header Content-Type is set to application/json

### **HTTP Error Code 500 Internal Server Error**

```
{
  "id":null,
  "status":"NOK",
  "messages":[
    "An unexpected error occurred"
  ],
  "additionalProperties":{

}
}
```

Please contact your technical support team

## **Process to create a recurring invoice for a new client:**

The following is the process to create a recurring invoice from scratch.

1. Add Client. Client id will be returned in the response.
2. Add Contact (optional). Pass the client id. Contact id will be returned in the response.
3. Add Payment Method. Pass client id. Payment Method id will be returned in the response.
4. Add Contract. Pass the client id and payment method id. Contract id will be returned in the response.
5. Add Recurring Invoice. Pass the client id, contact id (optional), contract id. Recurring invoice id is returned in response.

## Sample Java code:

```
public void testAddClient() {
    DefaultHttpClient httpClient = new DefaultHttpClient();
    try {

        URI url = new
URIBuilder().setPath("https://secure.billdozer.com/api/rest/client/add").build();

        HttpPost httpPost = new HttpPost(url);

        httpPost.setHeader("apikey", "xyz");
        httpPost.setHeader("password", "xyz");

        StringEntity input = new StringEntity("{json formatted request}");
        input.setContentType(MediaType.APPLICATION_JSON);
        httpPost.setEntity(input);

        HttpResponse httpResponse = httpClient.execute(httpPost);
        if (httpResponse.getStatusLine().getStatusCode() != 200) {
            result = httpResponse.getStatusLine().getStatusCode() + " " +
httpResponse.getStatusLine().getReasonPhrase();
            HttpEntity httpEntity = httpResponse.getEntity();
            result = result + ": " + EntityUtils.toString(httpEntity);
            return;
        }

        HttpEntity httpEntity = httpResponse.getEntity();
        result = EntityUtils.toString(httpEntity);

        ObjectMapper objectMapper = new ObjectMapper();
        com.billdozer.model.api.Response response =
objectMapper.readValue(result, com.billdozer.model.api.Response.class);

        System.out.println(response.toString());

    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        httpClient.getConnectionManager().shutdown();
    }
}
```