



COMMON

BadGatewayException

Kind: Class

VERSION
47

UPDATED
7 August 2026

DOCUMENT

reference-subsystem-packages-common/class-badgatewayexception

CONTENTS

1	Diagram	2	3	AI Coding Instructions	3
2	Usage	2			

Kind: Class

Source: `packages/common/exceptions/bad-gateway.exception.ts`

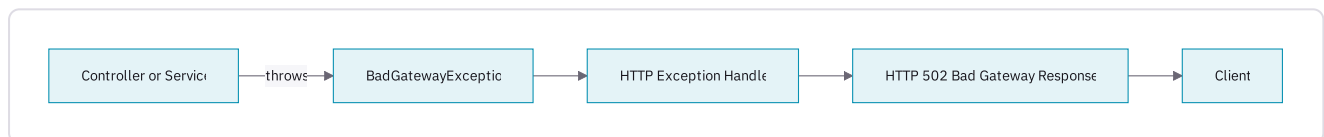
Part of: Common

Defines an HTTP exception for *Bad Gateway* type errors.

`BadGatewayException` represents an HTTP 502 Bad Gateway error, typically used when an upstream service, proxy, or gateway returns an invalid response. It extends the framework's HTTP exception handling so the error is converted into a consistent HTTP response for clients.

Extends: `HttpException`

1 Diagram



2 Usage

```
import { BadGatewayException, Injectable } from '@nestjs/common';

@Injectable()
export class PaymentService {
  async chargeCustomer() {
    const response = await fetch('https://payments.example.com/charge');

    if (!response.ok) {
      throw new BadGatewayException(
        'The payment provider returned an invalid response.',
      );
    }

    return response.json();
  }
}
```

3 AI Coding Instructions

- Throw `BadGatewayException` when a failure originates from an upstream dependency, such as an external API, reverse proxy, or service gateway.
- Do not use this exception for client validation failures; use appropriate 4xx exceptions for invalid requests or authorization issues.
- Provide a client-safe error message and avoid exposing upstream response bodies, credentials, or internal infrastructure details.
- Allow the framework's global exception filter to serialize the exception into a standard HTTP 502 response.