

# LU06.L01 - Autorisierung

[authorization.py](#)

```
from functools import wraps

import jwt
from flask import request, make_response, current_app, jsonify, g

def valid_token(func):
    """
    checks if the authorization token is valid
    :param func: callback function
    :return:
    """

    @wraps(func)
    def decorator(*args, **kwargs):
        if 'Authorization' in request.headers:
            token = request.headers['Authorization']
            try:
                data = jwt.decode(token[7:],
current_app.config['ACCESS_TOKEN_KEY'], algorithms=["HS256"])
                g.userrole = data['userrole']
            except Exception:
                return make_response('Invalid token!', 401)

            return func(*args, **kwargs)
        else:
            return make_response('token is missing!', 401)

    return decorator

def customer_required(func):
    """
    checks if the userrole is 'customer' or higher
    :param func: callback function
    :return:
    """

    @wraps(func)
    def decorator(*args, **kwargs):
        if 'userrole' in g:
            if g.userrole in ['customer', 'employee', 'admin']:
                return func(*args, **kwargs)
            return make_response('You shall not pass!', 401)
        return decorator
```

```
def employee_required(func):
    """
    checks if the userrole is 'employee' or higher
    :param func: callback function
    :return:
    """

    @wraps(func)
    def decorator(*args, **kwargs):
        if 'userrole' in g:
            if g.userrole in ['employee', 'admin']:
                return func(*args, **kwargs)
            return make_response('You shall not pass!', 401)
        return decorator

def admin_required(func):
    """
    checks if the userrole is 'admin'
    :param func: callback function
    :return:
    """

    @wraps(func)
    def decorator(*args, **kwargs):
        if 'userrole' in g:
            if g.userrole == 'admin':
                return func(*args, **kwargs)
            return make_response('You shall not pass!', 401)
        return decorator
```

## M321-LU06



Marcel Suter

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/modul/m321/learningunits/lu06/loesungen/autorisation>

Last update: **2025/02/24 08:05**

