HOCHSCHULE FULDA
UNIVERSITY OF APPLIED SCIENCES

WEB-APPLIKATIONEN

MVC 2: CONTROLLER

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

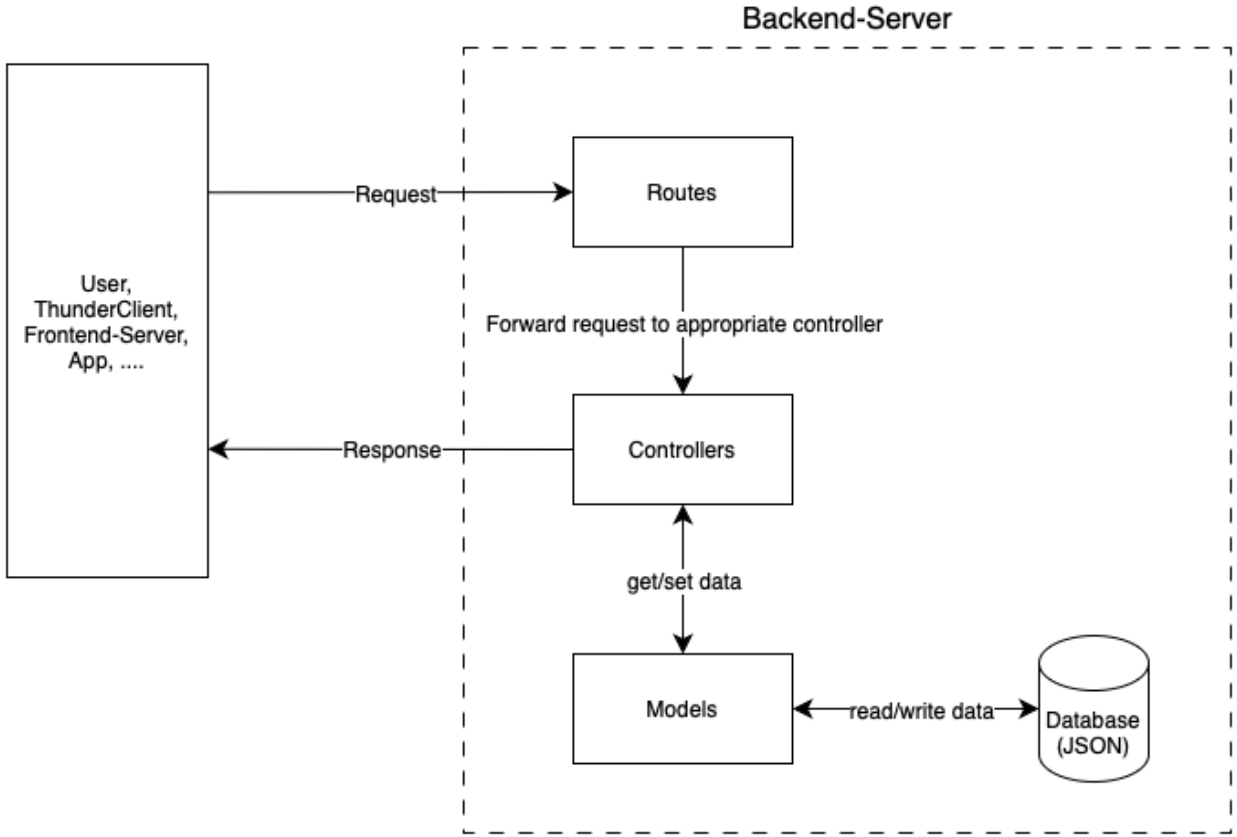
MVC

1. MODEL VIEW CONTROLLER (MVC)
2. (ENTWURFS-) MUSTER ZUR UNTERTEILUNG VON SOFTWARE
3. BESTANDTEILE:
    - Datenmodell (Model)
    - Präsentation (View)
    - Programmsteuerung (Controller)
4. WIEDERVERWENDBARKEIT DER EINZELNEN KOMPONENTEN

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

STRUKTUR DES BACKEND-SERVERS

1. BACKEND IMPLEMENTIERT
   - › Routes
   - › Controllers
   - › Models
2. BEINHALTET EINEN TEIL DES MVC-PATTERNS



AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

CONTROLLER

1. „ROUTEN“ LEITEN ANFRAGEN AN DEN ENTSPRECHENDEN CONTROLLER WEITER
2. CONTROLLER
   - Holen Daten aus dem Model
   - Aktualisieren das Model
   - Senden Response an den Client oder die Route zurück

VON ROUTEN ZU CONTROLLERN

1. CONTROLLER LIEST DATEN DES MODELS EIN
2. ENTHÄLT FUNKTIONEN UND RÜCKGABEN AN CLIENT ODER DIE ROUTE
3. `module.exports` STELLT FUNKTIONEN BEREIT

```
var database = {
    users: require('../models/users.json'),
    setUser: function (data) { this.users = data }
};

const getAllUsers = (req, res) => {
    res.json(database.users);
}

const createUser = async (req, res) => {···
}

module.exports = {
    getAllUsers,
    createUser
}
```

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

VON ROUTEN ZU CONTROLLERN

› ROUTE BINDET CONTROLLER EIN

› RUFT ENTSPRECHENDE FUNKTIONEN DES CONTROLLERS AUF

```
var express = require('express');
var router = express.Router();
var usersController = require('../controllers/userController')

/* GET users listing. */
router.route('/')
  .get(usersController.getAllUsers)
  .post(usersController.createUser);

module.exports = router;
```

MVC UND ROUTEN

1. ROUTEN BINDEN CONTROLLER EIN UND LEITEN ANFRAGEN AN DIESEN WEITER
2. MODEL ENTHÄLT DATEN
3. VIEW ENTHÄLT DARSTELLUNG
4. CONTROLLER
   - lesen/schreiben in Model
   - Erstellen Rückgabe an Client
5. RÜCKGABE KANN HTML SEIN, MUSS ES ABER NICHT
   - res.send
   - res.json
   - res.render
   - return
   - ...

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

HOCHSCHULE FULDA
UNIVERSITY OF APPLIED SCIENCES

WEB-PROGRAMMIERUNG

MIDDLEWARE

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

MIDDLEWARE

Middleware ist ein (lose definierter) Begriff für jede Software oder Service, die es den Teilen eines Systems ermöglicht, miteinander zu kommunizieren und Daten zu verwalten. Es ist die Software, die die Kommunikation zwischen den Komponenten und die Ein- und Ausgabe handhabt, so dass sich die Entwickler auf den spezifischen Zweck ihrer Anwendung konzentrieren können.

In server-seitigen Webanwendungs-Frameworks wird der Begriff oft spezifischer verwendet, um sich auf vorgefertigte Softwarekomponenten zu beziehen, die der Anfrage/Antwort-Verarbeitungspipeline des Frameworks hinzugefügt werden können, um Aufgaben wie den Datenbankzugriff zu erledigen.

https://developer.mozilla.org/de/docs/Glossary/Middleware

MIDDLEWARE IN WEBANWENDUNGEN

1. OFT VORGEFERTIGTE SOFTWAREKOMPONENTEN
2. KÖNNEN EINFACH DEM PROJEKT HINZUGEFÜGT WERDEN
3. ERWEITERN DIE FUNKTIONALITÄT DER ANWENDUNG
4. ERMÖGLICHEN FOKUSSIERUNG AUF DIE EIGENTLICHE ANWENDUNG

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

BEISPIEL: LOGGING

1. STANDARD IN EXPRESS:

   MORGAN

2. UNTERSCHIEDLICHE AUSGABEN
   - tiny
   - common
   - combined
   - etc.

```
16    app.use(logger('dev'));
17    app.use(express.json());
```

PROBLEME AUSGABE DEBUGGING-KONSOLE TERMINAL

```
manuel@MBP-von-Manuel middleware % npm start

> middleware@0.0.0 start
> node ./bin/www

GET / 200 14.875 ms - 207
GET /stylesheets/style.css 304 1.853 ms - -
```

LOGGING
LOGS IN DATEI SPEICHERN

```
var fs = require('fs')
var logger = require('morgan');

var indexRouter = require('./routes/index');
var usersRouter = require('./routes/users');

var app = express();

// Log to File
var accessLogStream = fs.createWriteStream(path.join(__dirname, 'access.log'), { flags: 'a' });
app.use(logger('combined', { stream: accessLogStream }));
```

AI
Manuel Richardt, Dominik Rupprecht
FB **ANGEWANDTE INFORMATIK**

ZUSAMMENFASSUNG

1. ROUTEN BINDEN CONTROLLER EIN
2. CONTROLLER NEHMEN ANFRAGEN ENTGEGEN, LADEN UND MODIFIZIEREN GGF. DIE DATEN DES MODELS
3. MIDDLEWARE ERWEITERT DIE FUNKTIONALITÄT VON SOFTWARE INDEM ES (MEIST VORGEFERTIGTE) ZUSÄTZLICHE KOMPONENTEN EINBINDET.