
attrecto
NEXT TECH DIGITAL SOLUTIONS
NodeJs-Express Backend.
ISO 9001Certified
FT FINANCIAL TIMES 1000 Europe's Fastest Growing Companies
50 Technology Fast 50 2016 CENTRAL EUROPE
500 Technology Fast 500 2016 EMEA

ExpressJs

- Gyors, egyszerű NodeJs alapú webes keretrendszer
- Alapvető funkcionalításokat valósít meg:
  - Routing
  - Middleware
  - Errorhandling

https://expressjs.com

ExpressJs - Routing

- Routing:

  router.METHOD(PATH, HANDLER)

  - **router** - express router instance
  - **method** - HTTP request method
  - **path** - a kérés útvonala
  - **handler** - egy funkció ami kezeli az adott route-ot

```
const express = require('express');
const router = express.Router();

router.get('/hello', (req, res, next) => {
  res.send(`Hello world!`);
});
```

ExpressJs - Routing

- Route paraméterek:
  - Route path:

    /users/:userId/books/:bookId
  - Request URL:

    http://localhost:3000/users/11/books/12
  - req.params:

    { "userId": "11", "bookId": "12" }

```
const express = require('express');
const router = express.Router();

router.get('/users/:userId/books/:bookId',
(req, res) => {
  const params = req.params;
  console.log(params)
  res.send(params)
});
```

ExpressJs - Middleware

- function (request, response, next)
- Bármit lehet futtatni benne
- Módosítani lehet a request és response objecteket
- A next segítségével bele lehet avatkozni a request-response cycle-be

```
const express = require('express');
const router = express.Router();

router.use((req, res, next) => {
  console.log('Time:', Date.now())
  next()
 });

router.use((req, res, next) => {
  if (valami) {
      const error = new Error('Internal
server error');
      error.status = 500;
      next(error);
  }
  next();
});
```

ExpressJs - Errorhandling

- Úgy különböztetjük meg hogy 4 argumentuma van:
  (err, request, response, next)
- next(error) esetén átugorja a többi handlert és egyből az error handlerbe ugrik

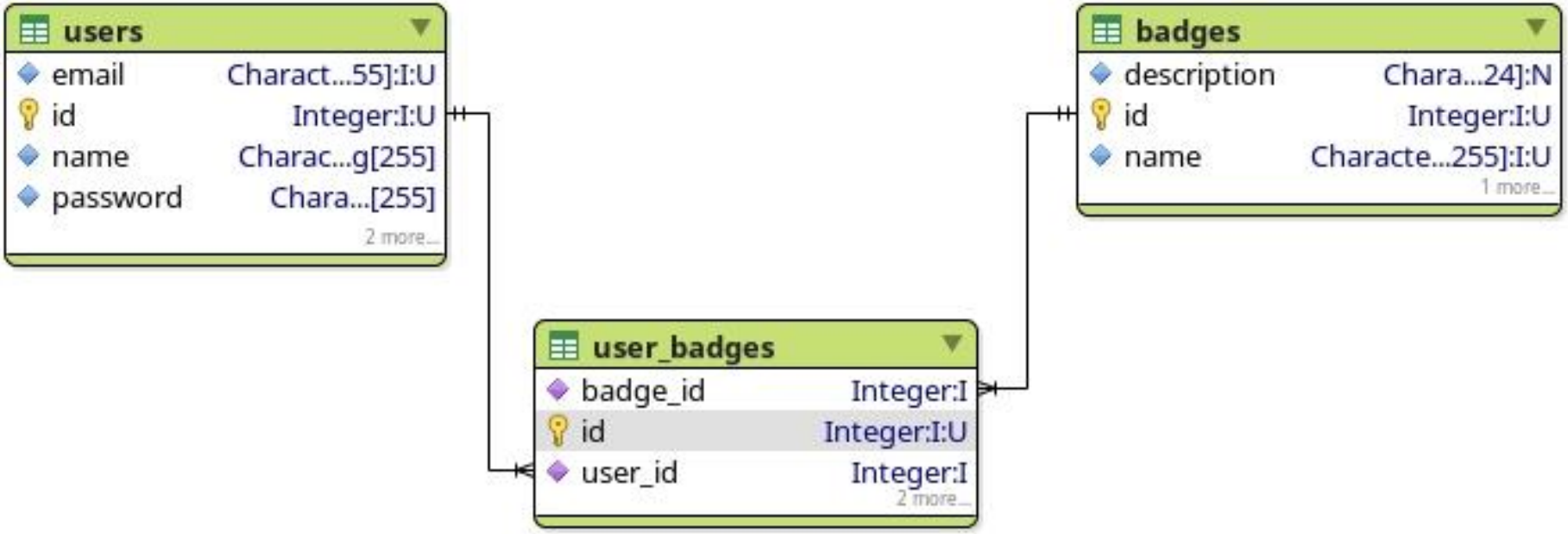
```
const express = require('express');
const app = express();

app.use(function (err, req, res, next) {
  console.error(err.stack)
  res.status(500).send('Something broke!')
});
```

Badge rendszer

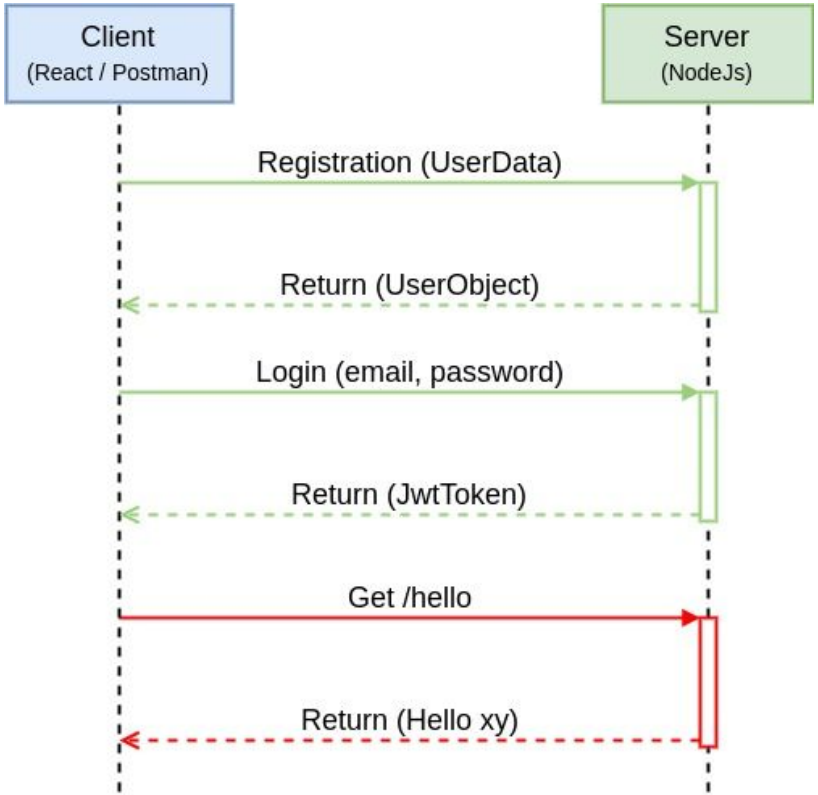
- Technológiák / Funkcionalítások
  - Adatbázis
    - File alapú (sqlite3)
  - User Regisztáció
    - password hash (bcryptjs)
  - Login
    - email és password validálás
    - token generálás (jsonwebtoken)
  - Middleware (Authentication)
    - token validálás

Adatbázis struktúra



Workflow

- User Registration:
  - Send UserData (email, password, ...)
  - Return CreatedUser (id, email, name)
- Login:
  - Send email + password
  - Return JwtToken
- Call an auth need WS:
  - Get /hello
  - Return ‘Hello xy’



Köszönöm a figyelmet!

Attrecto Zrt.
Attrecto Next Tech Digital Solutions

H-9024 Győr, Wesselényi str. 6.
info@attrecto.com