

1 Projekt 3: Heroku App

1.1 Wprowadzenie

Zmodyfikuj tak ostatni projekt, aby program używał 'Promises' bądź 'Async/Await' do wykonania asynchronicznego przepływu sterowania. Rozbuduj apkę o kolejne funkcje i następnie postaw ją na Heroku [<https://www.heroku.com/>]. Apka ma zapisywać każde logi do pliku (np w formie: data, żądanie HTTP, ścieżka). Do wykonywania żądań HTTP możecie użyć pakiet axios lub inny. Termin oddawania: niedziela 29 kwietnia 2018.

1.2 Ocenianie

Wykonanie poprawnie tego zadania jest na ocenę 4.0. Aby uzyskać ocenę wyższą należy wyróżnić się spośród grupy kreatywnością. Składa się na to przede wszystkim:

1. Pomysł na apke
2. Oryginalne rozwiązanie techniczne
3. Nawiązanie konstruktywnej (korzystnej dla obu stron) interakcji z zespołem developerów USOS API:
<https://usosapps.umk.pl/developers/api/contact/>
lub innym podmiotem związanym z portalem:
<https://usosapps.umk.pl/developers/>
w ramach zagadnień dotyczących projektu (proszę mnie dołączyć do korespondencji w CC).

1.3 Szablon: server-side

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

app.get('/', (req, res) => {

});
//dodaj obsługę logów
app.listen(port, () => {
  console.log(`server is running on port ${port}`);
});
```

1.4 Szablon: client-side

```
<html>
<body>
<h1>Nazwa apki</h1>
<script>
</script>
<footer>
  <p>Copyrights {{getYear}}</p>
</footer>
</body>
</html>
```

1.5 Zasada DRY

Kod ma być napisany z zastosowaniem zasady DRY (don't repeat yourself). <https://pl.wikipedia.org/wiki/DRY> Jeśli fragment kodu się powtarza to umieścić go w funkcji, obiekcie, pliku itp.