



# AKADEMIJA.IT

INFOBALT IR TECH CITY

## PRAKGINĖ UŽDUOTIS (PERLAIKYMAS)

Java Technologijos ir Priemonės

Andrius Stašauskas

andrius@stasauskas.lt

<http://stasauskas.lt/itpro2018/>

# LAIKAS

- Pradžia
  - 8:55 (8:00)
- Pabaiga
  - 12:25
- Patarimas: vėliausiai 12 valandą atlikite git commit ir git push į repozitoriją, kad liktų laiko, jeigu kažkas nepavyksta, net jeigu užduoties dar nepabaigėte
- Vėliau negu 12:25 įkelti pakeitimai nebus vertinami
- Užduotis tikrinimui bus paimama komanda git clone, leidžiamos komandos mvn eclipse:clean idea:clean, mvn clean install, startuojamas serveris

```
mvn clean install org.codehaus.cargo:cargo-maven2-plugin:1.7.0:run \  
-Dcargo.maven.containerId=tomcat8x -Dcargo.servlet.port=8081 \  
-Dcargo.maven.containerUrl=http://repol.maven.org/maven2/org/apache/tomcat/tomcat/8.5.35/tomcat-8.5.35.zip
```



## UŽDUOTIS (1/2)

- Sukurti įrašų studijų valdymo programą
  - egzistuoja galimybė pridėti įrašų studiją
- įrašų studijos atributai: pavadinimas, logotipas, kategorija, dydis
  - egzistuoja galimybė įrašų studiją susieti su atlikėju ir nuo jo atsieti
- įrašų studijos kategorija: valstybinė, namų, gyvai (live)
- įrašų studijos dydis: maža, vidutinė, didelė
- atlikėjo atributai: pavadinimas, tikras vardas, tikra pavardė, žanras, valstybė, gimimo data, nuotrauka
  - egzistuoja galimybė atlikėją susieti su daina ir nuo jo atsieti
- dainos atributai: pavadinimas, albumo pavadinimas, ilgis sekundėmis, mp3 failas



## UŽDUOTIS (2/2)

- Tas pats atlikėjas gali būti priskirtas skirtingoms studijoms
- Ta pati daina gali būti priskirta skirtingiems atlikėjams
- Įrašų studijos ir atlikėjo pavadinimai turi būti unikalūs
- Įrašų studjos peržiūroje prie kiekvieno atlikėjo matoma atlikėjo nuotrauka, matomas studijos logotipas
- Egzistuoja galimybė peržiūrėti visas įrašų studijas, visus atlikėjus ir visas dainas
- Egzistuoja galimybė pakeisti įrašų studijų, atlikėjų ir dainų atributus
- Egzistuoja galimybė pašalinti įrašų studiją, atlikėją ir dainą
- Egzistuoja galimybė ieškoti atlikėjo ir dainos



# PATEIKIMAS

- GitHub/BitBucket Public Repository
  - repository turi būti naujas
- Naudojamas Git
- Nuorodas į repozitoriją įkelti čia
  - <http://stasauskas.lt/linas>
  - darbų priėmimo statusas **live** : <http://stasauskas.lt/verti>
- Nuoroda turi veikti su git clone
- Projektas turi būti švarus
  - be target/build/node\_modules katalogų, be IDE projekto failų
- repozitorijoje turi būti tik **katalogai** kodas-java ir kodas-ui
  - kur kodas - jūsų kodas, įrašytas teste
  - kodas-java kataloge turi egzistuoti pom.xml failas
  - kodas-ui kataloge turi egzistuoti package.json failas



# STRUKTŪRA

- susideda iš 1 ( **vieno** ) maven projekto
  - kodas-java (war tipo)
- ir iš 1 ( **vieno** ) npm projekto
  - kodas-ui (npm tipo, axios turi veikti su 8081 portu)
- aplikacija turi būti kataloge kodas-java
- aplikacijos pavadinimas - kodas-java
- aplikacijos rezultatas - kodas-java.war failas
- aplikacijos paketas - com.studio.records
- aplikacija turi kompiliuotis be klaidų
- aplikacija neturi mesti išimčių naršyklėje ir konsolėje
- aplikaciją turi būti galima paleisti naudojant komandą:

```
mvn clean install org.codehaus.cargo:cargo-maven2-plugin:1.7.0:run \
-Dcargo.maven.containerId=tomcat8x -Dcargo.servlet.port=8081 \
-Dcargo.maven.containerUrl=http://repo1.maven.org/maven2/org/apache/tomcat/tomcat/8.5.35/tomcat-8.5.35.zip
```



## SERVISAI

- Turi būti naudojami tik Spring beans'ai
- bent vienas Spring bean turi būti aprašytas xml faile
- Turi egzistuoti bent vienas funkcionalus request apimties bean
- Servisai turi būti singleton apimties, bet juose negali būti saugomi duomenys
- Duomenys gali būti saugomi prototype/request bean (sąrašai, sąrašų elementai, kurie vėliau saugomi DB, ir pan.)
- Duomenimis dalintis tarp Rest ir Controller ir Servisų turi būti naudojamos skirtingos klasės negu dirbant su DB



## SERVISAI

- Visi sukurti bean'sai turi turėti logginimą į žurnalą (java, log4j, slf4j arba kitokį):
  - prieš susikurdami turi apie tai informuoti LOG žinute
  - prieš susinaikindami turi apie tai informuoti LOG žinute
  - žinutėje turi matytis bean name arba klasė ir apimtis (scope)
- System.out naudoti negalima





## DUOMENŲ BAZĖ (1/2)

- Duomenų struktūra turi tenkinti užduoties reikalavimus
- Įvesti duomenys turi išlikti aplikaciją netikėtai išjungus
- Įjungus turi užsikrauti prieš tai išsaugoti duomenys
- Aplikacija turi būti sukurta taip, kad veiktų tiek su tuščia, tiek su užpildyta duomenų baze
- Trinant atlikėją, kartu turi išsitrinti ir susietos dainos, jeigu jos priskirtos tik tam vienam atlikėjui



## DUOMENŲ BAZĖ (2/2)

- DB reikalavimai
  - turi būti panaudotas JPA darbui su DB
  - aplikacija turi būti sukurta taip, kad veiktų tiek su tuščia, tiek su užpildyta duomenų baze
  - atlikėjų/dainų filtravimas turi vykti per DB
  - turi būti parašyta bent viena sudėtinga DB SQL užklausa
  - trinant atlikėją, jo sąsajų turi nelikti ir prie dainos
  - atlikėjo dainos pagal albumą surūšiuotos DB lygyje



## NAVIGACIJA

- Svetainė turi būti 1 neatsinaujinantis naršyklės puslapis
- Navigacijai tarp puslapių naudojamas React Router
- Navigacija gali vykti tiek naudojantis nuorodomis, tiek mygtukais
- Pirminis puslapis, į kurį patenkama, turi būti įrašų studijų sąrašo puslapis
  - į jį patekti turi būti galima tiesiog atidarius aplikaciją naudojant tokią <http://localhost:8081/kodas-java> ar panašią nuorodą



## VAIZDAI (1/3)

- Aplikacija turi turėti bent tokius atskirus vaizdus (js):
  - įrašų studijų sąrašas
  - įrašų studijos peržiūros langas su atlikėjų sąrašu
    - ties atlikėju pavaizduotos 3 bet kokios jo dainos
  - įrašų studijos pridėjimo/redagavimo langas
  - atlikėjo bei dainos pridėjimo langai
  - atlikėjo langas su albumų sąrašu po kuriais pavaizduotos dainos



## VAIZDAI (2/3)

- Vaizdo (presentation) komponente turi būti tik:
  - pats komponento turinys
- Vaizdai neturi būti išdarkyti, informacija aiškiai matytis, elementai išdėstyti struktūriškai, ekrano ribose
- Vaizduose esančios formos turi leisti dirbti su visais duomenimis, esančiais duomenų bazėje



## VAIZDAI (3/3)

- Naudojami resursai negali būti išoriniai
  - viskas turi būti war faile
    - taip pat ir mp3 failas
- Vaizduose privalo būti prijungtas 1 CSS failas, spalvoms ir kitiems UI pakeitimams turi būti naudojamas šis failas, o vaizduose nustatomos CSS klasės
  - style atributai nenaudojami
  - style per JavaScript nenustatomas (išskyrus CSS klases)



## SĄRAŠŲ VAIZDAI (1/2)

- Duomenų sąrašai turi būti išvedami naudojant [Bootstrap cards](#)
- Jeigu atlikėjas neturi nė vienos dainos, rodomas įspėjamasis alert pranešimas [Bootstrap alert](#) su nuoroda į dainos pridėjimą tam atlikėjui. Jo neturi matytis, pridėjus bent 1 dainą (turi iškart pasislėpti) ir jeigu jų yra daugiau
- Duomenis trinant turi būti panaudotas [Bootstrap modal](#) ir pasirodyti dialogas su paklausimu, ar tikrai trinti duomenis



## SĄRAŠŲ VAIZDAI (2/2)

- Prie kiekvienos dainos turi būti mp3 failas kurį turi būti galima paleisti
- panaudoti HTML5 <audio>
- turi matytis play/pause mygtukas su galimybe dainą išklausyti
- mp3 failas visoms dainoms gali būti tas pats





## REDAGAVIMO VAIZDAI (1/2)

- Studijos redagavimo lange matomi egzistuojantys atlikėjai
  - jame turi matytis visi aplikacijoje esantys atlikėjai
  - naujus atlikėjus galima pridėti ir jie atsiras šioje vietoje
- Atlikėjus reikia išrinkti iš DB atlikėjų lentelės, jie turi būti surūšiuoti DB lygyje pagal abėcėlę A-Z
- Atlikėjai priskiriami prie studijos mygtuko paspaudimu
- Dainos priskiriamos prie atlikėjo mygtuko paspaudimu
- Redagavimo vaizduose esantys laukai turi validuotis, spalvinimui naudoti [Bootstrap validation](#):
  - validuojama, ar laukas privalomas
  - validuojamas lauko dydis
  - validuojama, ar įvestas skaičius



## REDAGAVIMO VAIZDAI (2/2)

- Redagavimo languose formoms turi būti naudojamas [Bootstrap Form CSS](#)
- Formų laukai ir jų pavadinimai turi tarpusavyje lygiuotis
- Formoje turi būti kuo mažiau kodo (formos laukai ir jų išdėstymas)
  - visas puslapis neturi būti vienoje formoje
- Studijos pridėjimo formoje studijos tipą galima pasirinkti HTML Select
- Šalinant atlikėją, jeigu jis priskirtas daugiau negu 1 studijai, trynimo pranešime turi būti parodytas įspėjimas, su kuriomis studijomis atlikėjas yra susietas



## REST REIKALAVIMAI

- Rest servisuose turi būti galima:
  - CRUD operacijos su studija, atlikėju ir daina
  - gauti tiek sąrašą, tiek vieną studiją, atlikėją ar dainą
  - atlikėjo priskyrimo studijai operacija
  - dainos priskyrimo atlikėjui operacija
  - atlikėjo ir dainų filtravimas pagal pavadinimą
  - dainos paieška pagal atlikėjo pavadinimą
  - turi būti pasiekiamas Swagger per /swagger-ui.html



# SĒKMĒS



**AKADEMIJA.IT**

INFOBALT IR TECH CITY