TTÜ

**Projekti aines „Veebiteenused“, IDU0075**

**Dokumentatsioon**

Stanislav Jeršov 155251IAPB

IAPB52

Juhendaja: Tarvo Treier

Tallinn

2017

**Sisukord**

Sissejuhatus..................................................................................................................3

SOAP API.......................................................................................................................4

REST API......................................................................................................................13

Testimine....................................................................................................................19

Kokkuvõte...................................................................................................................20

**Sissejuhatus**

Antud dokumentatsioon kirjeldab projekti, mis oli tehtud aine „Veebiteenused“ raames.

Projekti eesmärgiks on filmide vaatamislisti veebiteenuse kirjeldamine, realiseerimine, kasutamine ja testimine.

Antud teenuse pakkujaks võiks olla veebisait, kus on võimalik filme online vaadata. Kasutaja võiks luua erinevaid watchlist`e ning lisada sinna filme, mis ta tahaks tulevikus vaadata või, näiteks, luua listi “Favourite Movies”, et kõik meeldivad filmid oleksid ühes kohas.

Tulemuseks on filmide vaatamislisti jaoks realiseeritud SOAP ja REST API, on loodud SOAPUI-ga tetsid mõlema API jaoks ning on loodud lihtne kliendirakendus, millega on võimalik osaliselt SOAP API funktsionaalsust katsetada.

Dokumentatsioon on mõeldud teenuse liideste kasutamise realiseerimise lihtsustamiseks ning liidese disaini põhimõttete andmiseks.

**SOAP API**

* addFilm
* getFilm
* addWatchlist
* getWatchlist
* getFilmList
* getWatchlistList
* addWatchlistFilm
* getWatchlistFilmList

Nüüd hakkab operatsioonide kirjeldamine.

**addFilm**

Operatsioon loob uue filmi, mille me võime tulevikus lisada vaatamislisti jne..

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

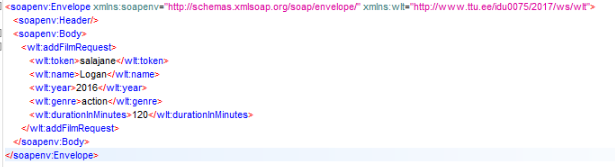
name – filmi nimi (String)

year – filmi aasta (BigInteger)

genre – filmi zanr (String)

durationInMinutes – filmi kestvus minutites (BigInteger)

Request:



Väljundandmed:

id – filmi unikaalne id, see on genereeritud automaatselt süsteemi poolt (BigInteger)

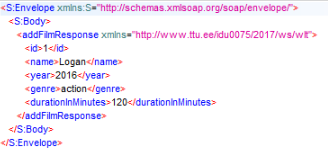
name – filmi nimi (String)

year – filmi aasta (BigInteger)

genre – filmi zanr (String)

durationInMinutes – filmi kestvus minutites (BigInteger)

Response:



**getFilm**

Operatsioon tagastab vastava filmi, kasutades etteantud filmi Id.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

id – filmi unikaalne id, see on genereeritud automaatselt süsteemi poolt (BigInteger)

Request:



Väljundandmed:

id – filmi unikaalne id, see on genereeritud automaatselt süsteemi poolt (BigInteger)

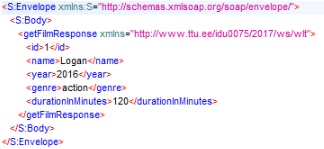
name – filmi nimi (String)

year – filmi aasta (BigInteger)

genre – filmi zanr (String)

durationInMinutes – filmi kestvus minutites (BigInteger)

Response:



**addWatchlist**

Operatsioon loob uue vaatamislisti. Vaatamislisti sisse võime panna filme, kasutades teist operatsiooni.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

name – filmi nimi (String)

Request:



Väljundandmed:

id – filmi unikaalne id, see on genereeritud automaatselt süsteemi poolt (BigInteger)

name – filmi nimi (String)

Response:



**getWatchlist**

Operatsioon tagastab vaatamislisti, kasutades etteantud vaatamislisti Id.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

id – filmi unikaalne id, see on genereeritud automaatselt süsteemi poolt (BigInteger)

Request:



Väljundandmed:

id – filmi unikaalne id, see on genereeritud automaatselt süsteemi poolt (BigInteger)

name – filmi nimi (String)

watchlistFilmList – vaatamislisti filmide nimekiri. Pärast vaatamislisti lisamist fimide list on tühi.

Response:



**getFilmList**

Operatsioon on kasutatud selleks, et kuvada kõike filme.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

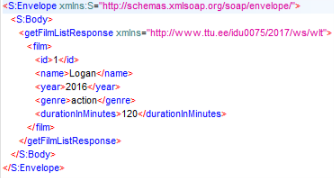
Request:



Väljundandmed:

filmList – kõikide lisatud filmide nimekiri (võib olla ka 0)

Response:



**getWatchlistList**

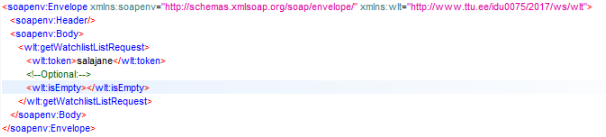
Operatsioon on kasutatud selleks, et kuvada kõike vaatamisliste.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

isEmpty – mittekohustuslik parameeter. Kui isEmpty väärtus on „jah“, siis tagastatakse ainult need vaatamislistid, mis on tühi. Kui isEmpty väärtus on „ei“, siis tagastatakse kõik need vaatamislistid, mis sisaldavad filme. Kui isEmpty väärtus null, siis tagastatakse kõiki vaatamisliste.

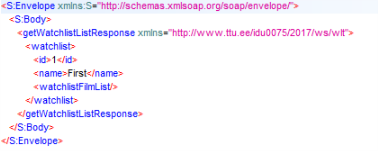
Request:



Väljundandmed:

watchlistList – vaatamislistide nimikiri (võib olla ka 0)

Response:



**addWatchlistFilm**

Operatsioon lisab filmi vaatamislisti.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

filmId – filmi id. Film selle Id-ga lisatakse vaatamislisti.

watchlistId – vaatamislisti id, kuhu kasutaja soovib lisada filmi.

Request:

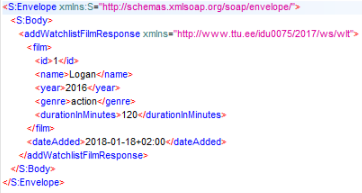


Väljundandmed:

film – lisatud film ja selle andmed

dateAdded – filmi vaatamislisti lisamise kuupäev.

Response:



**getWatchlistFilmList**

Operatsioon tagastab filmi listi, mis on etteantud Id-ga vaatamislisti sees.

Sisendandmed:

token – meie ülesanne raames see peab olema „salajane“ (String)

watchlistId – soovitud vaatamislisti Id.

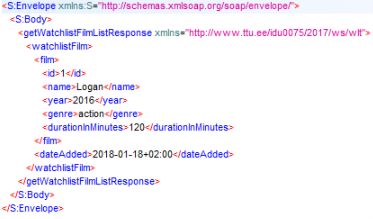
Request:



Väljundandmed:

Vaatamislisti kõiki filmide list.

Response:



**REST API**

* addFilm
* getFilm
* addWatchlist
* getWatchlist
* getFilmList
* getWatchlistList
* addWatchlistFilm
* getWatchlistFilmList

Nüüd hakkab operatsioonide kirjeldamine.

**addFilm**

Operatsioon loob uue filmi, mille me võime tulevikus lisada vaatamislisti jne..

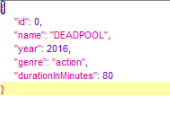
Method: POST

Ressurss: films

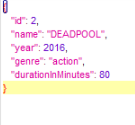
URL: /WatchlistWebApplication/webresources/films/?token=salajane

Request:

Id, mis me võime näha päringus ikkagi asendatakse süsteemi poolt genereeritud Id-ga.



Response:



**getFilm**

Operatsioon tagastab vastava filmi, kasutades etteantud filmi Id.

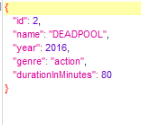
Method: GET

Ressurss: films

URL:

/WatchlistWebApplication/webresources/films/2?token=salajane , kus „2“ on filmi Id.

Response:



**addWatchlist**

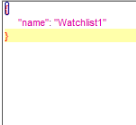
Operatsioon loob uue vaatamislisti. Vaatamislisti sisse võime panna filme, kasutades teist operatsiooni.

Method: POST

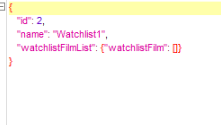
Ressurss: watchlists

URL: /WatchlistWebApplication/webresources/watchlists/?token=salajane

Request:



Response:



**getWatchlist**

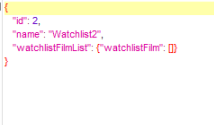
Operatsioon tagastab vaatamislisti, kasutades etteantud vaatamislisti Id.

Method: GET

Ressurss: watchlists

URL: /WatchlistWebApplication/webresources/watchlists/2?token=salajane , kus “2” on vaatamislisti Id.

Response:



**getFilmList**

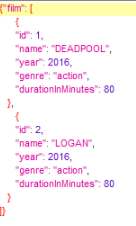
Operatsioon on kasutatud selleks, et kuvada kõike filme.

Method: GET

Ressurss: films

URL: /WatchlistWebApplication/webresources/films/?token=salajane

Response:



**getWatchlistList**

Operatsioon on kasutatud selleks, et kuvada kõike vaatamisliste.

Operatsiooni tulemuse võib filtreerida kasutades isEmpty parameetrit.

Method: GET

Ressurss: watchlists

Näidis URL: /WatchlistWebApplication/webresources/watchlists/?token=salajane&isEmpty=null , kus „null“ on parameetri isEmpty value (võib olla „jah“, „ei“ või „null“).

Response:



**addWatchlistFilm**

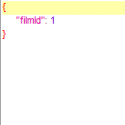
Operatsioon lisab filmi vaatamislisti.

Method: POST

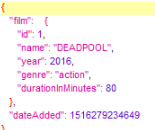
Ressurss: watchlists

URL: /WatchlistWebApplication/webresources/watchlists/1/films?token=salajane , kus “1” on vaatamislisti Id.

Request :



Response:



**getWatchlistFilmList**

Operatsioon tagastab filmi listi, mis on etteantud Id-ga vaatamislisti sees.

Method: GET

Ressurss: watchlists

URL: /WatchlistWebApplication/webresources/watchlists/2/films?token=salajane , kus „2“ on vaatamislisti Id.

Response:



**Testimine**

SOAP ja REST API jaoks olid loodud testid SoapUI-s.

SOAP API operatsioonide jaoks on kaks testi: „Contains“, mis kontrollib, kas saanud vastus sisaldab mingi oodatud andmed või mitte ja „Schema Compliance“, mis kontrollib vastavust schema`le.

REST API operatsioonide jaoks on üks test – „Contains“

**Kokkuvõte**

Tulemuseks on filmide vaatamislisti jaoks realiseeritud SOAP ja REST API, on loodud SOAPUI-ga tetsid mõlema API jaoks ning on loodud lihtne kliendirakendus, millega on võimalik osaliselt SOAP API funktsionaalsust katsetada.

Töö tegemise käigul ma tegelesin veebiteenuse kirjeldamisega, realiseerimisega, kasutamisega ja testimisega.

Kursuse käigul saanud teadmised olid kasulikud projekti tegemise jaoks ning projekt ise annab arusaamise kuidas veebiteenuste loomine ja sellega seotud teised tegevused välja näevad