Web service specification v1

Contents

[Current User Points 4](#_Toc465642999)

[Description 4](#_Toc465643000)

[Parameters 4](#_Toc465643001)

[Request example 4](#_Toc465643002)

[Structure 4](#_Toc465643003)

[Videos filtered by tags 5](#_Toc465643004)

[Description 5](#_Toc465643005)

[Parameters 5](#_Toc465643006)

[Request example 5](#_Toc465643007)

[Structure 5](#_Toc465643008)

[Video comments 6](#_Toc465643009)

[Description 6](#_Toc465643010)

[Parameters 6](#_Toc465643011)

[Request example 6](#_Toc465643012)

[Structure 6](#_Toc465643013)

[Add point and/or view 7](#_Toc465643014)

[Description 7](#_Toc465643015)

[Parameters 7](#_Toc465643016)

[Request example 7](#_Toc465643017)

[Structure 7](#_Toc465643018)

[Add comment to video 8](#_Toc465643019)

[Description 8](#_Toc465643020)

[Parameters 8](#_Toc465643021)

[Request example 8](#_Toc465643022)

[Structure 8](#_Toc465643023)

[Manage video like 9](#_Toc465643024)

[Description 9](#_Toc465643025)

[Parameters 9](#_Toc465643026)

[Request example 9](#_Toc465643027)

[Structure 9](#_Toc465643028)

[Update user information 10](#_Toc465643029)

[Description 10](#_Toc465643030)

[Parameters 10](#_Toc465643031)

[Request example 10](#_Toc465643032)

[Structure 10](#_Toc465643033)

[Log in user 11](#_Toc465643034)

[Description 11](#_Toc465643035)

[Parameters 11](#_Toc465643036)

[Request example 11](#_Toc465643037)

[Structure 11](#_Toc465643038)

[Login succeeded 11](#_Toc465643039)

[Login falied – login information incorrect, deleted user, blocked user, deactivated user (check each one) 11](#_Toc465643040)

[All geographic units 13](#_Toc465643041)

[Description 13](#_Toc465643042)

[Parameters 13](#_Toc465643043)

[Request example 13](#_Toc465643044)

[Structure 13](#_Toc465643045)

[Register user to app 14](#_Toc465643046)

[Description 14](#_Toc465643047)

[Parameters 14](#_Toc465643048)

[Request example 14](#_Toc465643049)

[Structure 14](#_Toc465643050)

[Change password 15](#_Toc465643051)

[Description 15](#_Toc465643052)

[Parameters 15](#_Toc465643053)

[Request example 15](#_Toc465643054)

[Structure 15](#_Toc465643055)

[Deactivate account 16](#_Toc465643056)

[Description 16](#_Toc465643057)

[Parameters 16](#_Toc465643058)

[Request example 16](#_Toc465643059)

[Structure 16](#_Toc465643060)

[Activate account (not meant to be web service!!!! Change to web page for production... ) 17](#_Toc465643061)

[Description 17](#_Toc465643062)

[Parameters 17](#_Toc465643063)

[Request example 17](#_Toc465643064)

[Structure 17](#_Toc465643065)

[Change telecommunication operator 18](#_Toc465643066)

[Description 18](#_Toc465643067)

[Parameters 18](#_Toc465643068)

[Request example 18](#_Toc465643069)

[Structure 18](#_Toc465643070)

# Current User Points

## Description

Web service returns current user points for telecommunication operator to use that information for discount and benefits for user in their webshop.

## Parameters

user\_id – active user's id currently logged in

telecomm\_operator – active telecomm operator for logged-in user

## Request example

GET localhost/webservice/current\_user\_points?user\_id=1&telecomm\_operator=2

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"user\_id\_remote":"<user\_id\_remote>",

"points": ##

}

}

# Videos filtered by tags

## Description

This webservice returns videos and likes for user to choose which video he wants to watch. Those videos are filtered by tags.

## Parameters

gender - logged-in user's gender

birthday/age – logged-in user's birthday date or calculated age (dvoumim se)

geographic\_unit – logged-in user's geographic unit

## Request example

GET localhost/webservice/videos?gender=M&birthday=1992.20.10&geographic\_unit=2

GET localhost/websevice/videos?gender=M&age=23&geographic\_unit=2

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

[

{

"id":###,

"title":"<title>",

"link":"<link>",

"likes":###,

"dislikes":###,

"points":###,

"created\_at":"<created\_at>"

},

{

"id":###,

"title":"<title>",

"link":"<link>",

"likes":###,

"dislikes":###,

"points":###,

"created\_at":"<created\_at>"

}

]

}

}

# Video comments

## Description

Web service which returns all comments for video that is watched at least one time.

## Parameters

video – watched video's id

## Request example

GET localhost/webservice/video\_comments?video=1

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

[

{

"id":###,

"author":{

"id":"<id>",

"name":"<name>",

"surname":"<surname>"

},

"content":"<content>",

"created\_at":"date and time of creation"

},

{

"id":###,

"author":{

"id":"<id>",

"name":"<name>",

"surname":"<surname>"

},

"content":"<content>",

"created\_at":"date and time of creation"

}

]

}

}

# Add point and/or view

## Description

Web service adds point and/or view. If user hasn't seen the video yet, points are added to user and video view too. If video has one view, extra view is added to user's video views.

## Parameters

video – video's id that is seen

user – user's id that is logged-in currenty

## Request example

GET localhost/webservice/view\_video?video=1&user=3

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Add comment to video

## Description

Web service adds new comment to video that is currently open.

## Parameters

video – video's id that is currently open

author – user's id that wrote comment

content – comment's content that shall be added

## Request example

GET localhost/webservice/comment\_video?video=1&author=1&content=This is my comment

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Manage video like

## Description

Web service that updates user's like status.

## Parameters

video – video's id that is seen

user – user's id that is logged-in currenty

like - current like value

## Request example

GET localhost/webservice/videolike?video=1&user=1&like=-1

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Update user information

## Description

Web service to update user information

## Parameters

id – user's id that is logged-in currenty

name - user's name

surname – user's surname

birthday – user's birthday

gender – user's gender

email – user's email

username – user's username

geo\_unit – geographic unit's id

## Request example

POST localhost/webservice/updateuser with all parameters except password

NOTICE: use GET method to test web service , POST IS REQUIRED

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Log in user

## Description

Web service that checks login info for user and returns its information.

## Parameters

username – username for user to be logged in

password - hashed password for user to be logged in

## Request example

POST localhost/webservice/login?username=username&password=password

## Structure

### Login succeeded

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message":"<message>",

"user":{

"id":##,

"name": "<name>",

"surname": "<surname>",

"birthday": "<birthday>",

"gender": "<gender>",

"email": "<email>",

"username": "<username>",

"password": "<password>",

"password\_changed\_at": "<password\_changed\_at>",

"geographic\_unit": {

"id": ###,

"name": "<name>"

}

}

}

}

### Login falied – login information incorrect, deleted user, blocked user, deactivated user (check each one)

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# All geographic units

## Description

Web service returns all geographic units.

## Parameters

None

## Request example

GET localhost/webservice/geographic\_units

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":[

{

"id":###,

"name": "<name>"

},

{

"id":###,

"name": "<name>"

}

]

}

# Register user to app

## Description

Web service for registering users to application.

## Parameters

name - user's name

surname – user's surname

birthday – user's birthday

gender – user's gender

email – user's email

username – user's username

password – user's password

geo\_unit – geographic unit's id

## Request example

GET localhost/webservice/registration with all parameters specified

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Change password

## Description

Web service for changing user password. The web service returns false if password is equal to old one.

## Parameters

password – new user's password

id – user who wants to change password

## Request example

GET localhost/webservice/change\_password?id=1&password=nekalozinka

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Deactivate account

## Description

Web service for deactivating user account. If user is blocked, deactivated or deleted, web service returns false .

## Parameters

id – user id

## Request example

GET localhost/webservice/deactivate?id=1

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Activate account (not meant to be web service!!!! Change to web page for production... )

## Description

Web service for activating user account. If user is blocked, activated or deleted, web service returns false .

## Parameters

id – user id

## Request example

GET localhost/webservice/activate\_user?id=1

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}

# Change telecommunication operator

## Description

Web service for changing active telecommunication operator. If new one is the same as old one it returns error.

## Parameters

id – user id

telecomm – telecommunication operator id

## Request example

GET localhost/webservice/change\_telecomm?id=1&telecomm=2 (ODBAČENO)

GET localhost/webservice/user/1/telecomm/1/change

## Structure

{

"status":100,

"requested\_at":"<date and time>",

"data":{

"success":##,

"message": "<message>"

}

}