**Specifikacija web servisa**

1. Users\_controller

1.1 login\_post

* + logs user
  + params: JSON {username,password}
  + method post
  + https
  + return JSON {success}

1.2 logout

* + logs user out
  + method post
  + params JSON {user\_id}
  + returns JSON {success}

1.3 ban

* + bans user
  + requires admin
  + path “/users/[user\_id]/ban”
  + param user\_id
  + method get
  + returns JSON {success}

1.4 unban

* + unbans user
  + requires admin
  + path “/users/[user\_id]/unban”
  + param user\_id
  + method get
  + returns JSON {success}

1.5 edit

* + gets user data for editing (for admin)
  + requires admin
  + path “/users/[user\_id]/edit
  + param user\_id
  + method get
  + returns JSON {User}

1.6 edit\_post

* + sets edited data
  + path “/users/[user\_id]/edit
  + param user\_id
  + method post
  + returns JSON {User ; success}

1.7 get\_users

* + gets list of users
  + requires admin
  + path “/users”
  + method get
  + returns JSON{ Users\_List}

1.8. search\_users

* + get data for searching users
  + path "/users/search"
  + method post

1.9. get\_searched\_users

* + get list of searched users
  + path "/users/search\_list"
  + method get
  + return JSON { Searched\_users\_list ; success }

1.10 promote

* + promotes user to admin
  + requires admin
  + path “/users/[user\_id]/promote
  + method get
  + returns JSON {success}

1.11 demote

* + demotes user from admin
  + requires admin
  + path “/users/[user\_id]/demote
  + method get
  + returns JSON {success}

1. User\_Controller

2.1 profile

* + gets user data for editing (for user)
  + path “/users/profile
  + method get
  + returns JSON {User}

2.1 profile\_post

* + sets edited user data )
  + path “/users/profile
  + param JSON { edited\_data }
  + method post
  + returns JSON {User, success}

2.2 register

* + registers user
  + path “/user/register”
  + param JSON { username, email, password }
  + https
  + method post
  + returns JSON { success} , send email with registration\_hash\_token

2.3 verify

* + verify user registration
  + path “/user/verify”
  + param JSON { registration\_hash\_token}
  + method post
  + returns JSON {success}

2.4 decline

* + declines user registration
  + path “/user/decline”
  + param JSON { registration\_hash\_token}
  + method post
  + returns JSON {success}

2.5 reset

* + resets user password
  + path “/user/reset”
  + param JSON {username} or JSON {email}
  + method post
  + returns JSON {success}, sends email with reset\_hash\_token

2.6 reset\_confirm

* + confirms password reset
  + path”user/reset\_confirm”
  + param JSON{reset\_hash\_token}
  + method post
  + returns JSON {success}, sends email with new password

2.7 reset\_cancel

* + cancels password reset
  + path “/user/reset\_cancel”
  + param JSON{reset\_hash\_token}
  + method post
  + returns JSON {success}

1. AD\_Controller

3.1 create\_ad

* + creates new ad
  + path “ads”
  + param JSON{/ad\_data}
  + method post
  + returns JSON{ ad, success}

3.2 delete\_ad

* + deletes existing ad
  + requires admin
  + path “/ads/[ad\_id]/delete”
  + param ad\_id
  + method get
  + returns JSON { success}

3.3 update\_ad

* + updates existing ad
  + path “/ads/[ad\_id]/update”
  + param ad\_id, JSON{ data\_to\_update}
  + method get
  + returns JSON { ad, success}

3.4 get\_ad

* + get specified ad
  + path “/ads/[ad\_id]”
  + param ad\_id
  + method get
  + returns JSON { ad, success}

3.5 get\_ads

* + gets all ads
  + path “/ads”
  + method get
  + returns JSON { ad\_List, success}

3.6 search\_ads

* + search ads, returns list of ads according to parameters
  + path “/ads/search”
  + method get
  + returns JSON {ad\_List, success}

1. Message\_controller

4.1 create\_message

* + creates new message
  + path “/messages”
  + param JSON { message\_data }
  + method post
  + returns JSON{success}

4.2 delete\_message

* + deletes existing message
  + path “/messages/[message\_id]/delete”
  + param message\_id
  + method post
  + returns JSON{success}

4.3 get\_recieved

* + gets all recieved messages
  + path “/messages/recieved”
  + method get
  + returns JSON {list\_recieved\_messages}

4.4 get\_sent

* + gets all sent messages
  + path “/messages/sent”
  + method get
  + returns JSON {list\_sent\_messages}

4.5 get\_messages

* + gets all messages
  + path “/messages/”
  + method get
  + returns JSON {list\_messages}

4.6 get\_message

* + gets specific message
  + path “/messages/[message\_id]/”
  + params message\_id
  + method post
  + returns JSON{success}

1. Ratings\_Controller

5.1 upvote\_user

* + upvotes rating for existing user
  + path “/users/[user\_id]”
  + param user\_id
  + method get
  + returns JSON{ success}

5.2 downvote\_user

* + downvotes rating for existing user
  + path “/users/[user\_id]/downvote”
  + param user\_id
  + method get
  + returns JSON{ success}

5.3. upvote\_ad

* + path “/ads/[ad\_id]/upvote"
  + params ad\_id
  + method get
  + returns JSON {success}

5.4. downvote\_ad

* + path “/ads/[ad\_id]/downvote"
  + params ad\_id
  + method get
  + returns JSON {success}

1. Transaction\_controller

6.1. create\_new

* + path "/transactions/new"
  + params user1\_id,user2\_id
  + method post
  + returns JSON {success}

6.2. get\_transactions

* + path "/user/[user\_id]/transactions"
  + params user\_id
  + method get
  + returns JSON {transactions}