Dokumentacja „chat-project”

[Frontend 1](#_Toc1151861684)

[Backend 1](#_Toc1202854114)

[Endpoints 4](#_Toc143897763)

# Frontend

getStringDate(date)

/\*\*

\* @author Rupert Rybka

\* @param {string} date – date to be converted

\* @returns {string} date in format *HH:MM YYYY-MM-DD*

\*/

getInitials(nick)

/\*\*

\* @author Rupert Rybka

\* @param {string} nick – nick from which initials will be extracted

\* @returns {string} initials of Nick (2 letters)

\*/

randomColor(n)

/\*\*

\* @author Rupert Rybka

\* @param {number} n – seed for color generator

\* @returns {string} random color in format *rgb(r,g,b)*

\*/

# Backend

register(req, res)

/\*\*

\* Function that is responsible for inserting user credentials to database, that is creating a new user

\* @author Rupert Rybka

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

login(req, res)

/\*\*

\* Function that is responsible for verifying access of user to application

\* @author Wiktor Kaszuba

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

sendMessage(req, res)

/\*\*

\* Function that is responsible for inserting a message to database

\* @author Wiktor Kaszuba

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

getChat(req, res)

/\*\*

\* Function that is responsible for sending the conversation details to logged user

\* @author Wiktor Kaszuba

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

getNick(req, res)

/\*\*

\* Function that sends nick of the current user of application

\* @author Rupert Rybka

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

getUsers(req, res)

/\*\*

\* Function that sends list of users, except for current user of application

\* @author Wiktor Kaszuba

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

forwardMessage(req, res)

/\*\*

\* Function that inserts to database information about message that is supposed to be forwarded

\* @author Wiktor Kaszuba

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

deleteMessage(req, res)

/\*\*

\* Function that deletes message from database

\* @author Wiktor Kaszuba

\* @param {RequestObject} req – handle to http request

\* @param {ResponseObject} res – handle to http response

\* @returns {Void}

\*/

# Endpoints

* backend/login (POST)
* backend/getUsers (POST)
* backend/getChat (POST)
* backend/register (POST)
* backend/sendMessage (POST)
* backend/deleteMessage (DELETE)
* backend/forwardMessage (POST)
* backend/getNick (POST)