**ДОДАТКИ**

**ДОДАТОК А**

**Код серверної частини програми**

var express = require('express');

var app = express();

var server = require('http').createServer(app);

var io = require('socket.io').listen(server);

var swig = require('swig');

var DB = require('./DBController');

var crypto = require('crypto');

var bodyParser = require('body-parser');

var cookieParser = require('cookie-parser');

var session = require('express-session');

var multipart = require('connect-multiparty');

var path = require('path');

var fs = require('fs');

var validator = require('validator');

var multipartMiddleware = multipart({ uploadDir: \_\_dirname + '/uploads' });

app.use(bodyParser());

app.use(cookieParser('secret-key-1199234'));

app.use(session());

app.use(express.static(\_\_dirname + '/public', { maxAge: 86400000 }));

app.engine('html', swig.renderFile);

app.set('view engine', 'html');

app.set('views', \_\_dirname + '/public');

app.set('view cache', false);

swig.setDefaults({ cache: false });

app.get('/', checkAuth, function (req, res) {

res.render('index-chat', { username : req.session.user\_name, userimage: req.cookies.userimage});

});

app.get('/login', function (req, res) {

res.render('login', { });

});

app.get('/login/error', function (req, res) {

res.render('login', { alert : ' <div class="alert alert-danger"><b>User name or password is incorrect</b></div>'});

});

app.post('/login/error', function (req, res) {

res.render('login', { alert : ' <div class="alert alert-danger"><b>User name or password is incorrect</b></div>'});

продовження додатку А

});

app.get('/register', function (req, res) {

res.render('register', { title:"Create Account" });

});

app.get('/register/error/:id', function (req, res) {

var errmessage = '';

switch(req.params.id)

{

case '1':

errmessage = '<div class="alert alert-danger"><b>Please specify correct login and password</b></div>';

break;

case '2':

errmessage = '<div class="alert alert-danger"><b>User with this login now exists. Please select other login and try again</b></div>';

break;

case '3':

errmessage = '<div class="alert alert-danger"><b>No image selected or wrong image format. Use only [jpg], [gif], [png] file types</b></div>';

break;

}

res.render('register', { title:"Create Account", alert : errmessage });

});

app.post('/register', multipartMiddleware, function (req, res) {

var name = req.body.login;

var email = req.body.email;

var pwd = req.body.password;

var \_hash = getMD5('secret-'+name+'-'+pwd);

var \_error = '';

var userimage = req.files.imagemy;

var tempPath = userimage.path;

// if wrong name / password entered

if(name == '' || pwd == '')

{

fs.unlink(tempPath, function (err) {

if (err) throw err;

console.error("Wrong data, image was removed");

продовження додатку А

\_error = '1';

res.redirect('/register/error/'+\_error);

});

}

else

{

var ext = path.extname(req.files.imagemy.originalFilename).toLowerCase();

var targetPath = path.resolve('./public/avatars/avatar\_' + name + ext);

var query = DB.connection.query('SELECT \* FROM users WHERE name = ?', [name], function(err, rows, fields) {

if (err) throw err;

if (rows[0]) {

fs.unlink(tempPath, function (err) {

if (err) throw err;

console.error("Wrong data, image was removed");

\_error = '2';

res.redirect('/register/error/'+\_error);

});

} else {

//checking image

if (ext === '.png' || ext === '.jpg' || ext === '.jpeg' || ext === '.gif') {

fs.rename(tempPath, targetPath, function(err) {

if (err) throw err;

console.log("Upload completed!");

DB.createUser(name,email,pwd,\_hash,'avatar\_'+name+ext,res);

});

} else {

fs.unlink(tempPath, function(err) {

if (err) throw err;

console.error("Only image files are allowed!");

\_error = '3';

res.redirect('/register/error/'+\_error);

});

}

}

});

}

});

продовження додатку А

app.post('/login', function (req, res) {

var query = DB.connection.query('SELECT \* FROM users WHERE name = ? AND password = ?', [req.body.login, req.body.password], function(err, rows, fields) {

if (err) throw err;

if (rows[0]) {

req.session.user\_id = rows[0].id;

req.session.user\_hash = rows[0].hash;

req.session.user\_name = rows[0].name;

var uavatar = rows[0].picture;

if(uavatar == undefined) uavatar = 'img/avatar3.png';

else

uavatar = 'avatars/'+uavatar;

res.cookie('userimage', uavatar, { });

res.cookie('userhash', req.session.user\_hash, { });

res.redirect('/');

} else {

res.redirect('/login/error');

}

console.log(rows);

});

console.log(query.sql);

});

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Logout \*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

app.get('/logout', function (req, res) {

delete req.session.user\_id;

delete req.session.user\_hash;

delete req.session.user\_name;

res.clearCookie('userhash', { });

res.clearCookie('userimage', { });

res.redirect('/login');

});

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Checking Authorization \*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

function checkAuth(req, res, next) {

if (!req.session.user\_id || !req.session.user\_hash || !req.session.user\_name) {

res.redirect('/login');

продовження додатку А

} else {

var query = DB.connection.query('SELECT \* FROM users WHERE id = ? AND hash = ?', [req.session.user\_id, req.session.user\_hash], function(err, rows, fields) {

if (err) throw err;

if (rows[0]) {

next();

} else {

res.send('Error auth');

}

});

console.log(query.sql);

}

}

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* get MD5 Hash \*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

function getMD5(data) {

return crypto.createHash('md5').update(data).digest('hex');

}

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\* New connection handler \*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

io.sockets.on('connection', function (socket) {

//User joined to chat

socket.on('join', function(data) {

var query = DB.connection.query('SELECT \* FROM users WHERE hash = ?', [data.hash], function(err, rows, fields) {

if (err) throw err;

if (rows[0]) {

var uname = rows[0].name;

var uavatar = rows[0].picture;

if(uavatar == undefined) uavatar = 'img/avatar3.png';

else

uavatar = 'avatars/'+uavatar;

socket.set('nickname', uname, function () {

socket.set('picture', uavatar, function () {

socket.emit('identifymessage', { sender : "Server", name: uname, message: 'Welcome to the chat, <b>'+uname+'</b>!' });

продовження додатку А

socket.broadcast.emit('privatemessage', {sender: 'Server', person:'home', message: 'User <b>'+uname+'</b> joined to chat.'});

//Sending to all users info about online users

io.sockets.emit('userlist', {userlist: getUserList()});

});

});

} else {

socket.emit('privatemessage', { sender : "Server", person:'home', date: getCurrentDate(), message: 'Error: Bad authorization ...' });

}

});

});

//User disconnected

socket.on('disconnect', function() {

socket.get('nickname', function (err, name) {

if(err)

console.log('Invalid user disconnected');

else

{

io.sockets.emit('userlist', {userlist: getUserList()});

io.sockets.emit('userlogout', {nickname: name});

socket.broadcast.emit('privatemessage', {sender: 'Server', person:'home', message: 'User <b>'+name+'</b> has been disconnected from server.'});

}

});

});

//previuos messages

socket.on('getpreviousmessages', function (data) {

socket.get('nickname', function (err, name) {

if(err)

console.log('Invalid user');

else

{

if (data.target == 'home')

{

var query = DB.connection.query('SELECT name, text, date, \'home\' AS target FROM messages' +

' INNER JOIN users ON ( messages.sender\_id = users.id ) '+

' WHERE receiver\_id = ?'+

продовження додатку А

ORDER BY date DESC LIMIT 100', [0], function(err, rows, fields) {

console.log(query.sql);

if (err) throw err;

if (rows[0]) {

io.sockets.emit('previousmessages', {messages: rows});

}

});

} // if (data.target == 'home')

else

{

var query = DB.connection.query('SELECT name, text, date, ? AS target FROM messages'+

' INNER JOIN users ON ( messages.sender\_id = users.id AND'+

' (messages.sender\_id = (SELECT id FROM users WHERE name = ? LIMIT 1)'+

' AND messages.receiver\_id = (SELECT id FROM users WHERE name = ? LIMIT 1)'+

' OR messages.sender\_id = (SELECT id FROM users WHERE name = ? LIMIT 1)'+

' AND messages.receiver\_id = (SELECT id FROM users WHERE name = ? LIMIT 1)))'+

' ORDER BY date DESC LIMIT 100 ', [data.target, name, data.target, data.target, name], function(err, rows, fields) {

console.log(query.sql);

if (err) throw err;

if (rows[0]) {

io.sockets.emit('previousmessages', {messages: rows});

}

});

} // (data.target != 'home')

}

});

});

//Message to some user

продовження додатку А

socket.on('sendto', function (data) {

socket.get('nickname', function (err, name) {

if(err)

console.log('Invalid user');

else

{

sendTo(data.target, name, data.message, function(result){

if(result > 0)

{

if(result == 1)

socket.emit('privatemessage', { sender : name, person : data.target, message: validator.escape(data.message) });

DB.saveMessage(name, data.target, validator.escape(data.message), getCurrentDate());

}

});

}

});

});

});

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Returns list with nline users \*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

function getUserList()

{

var clients = io.sockets.clients();

var arr = [];

clients.forEach(function(entry) {

entry.get('nickname', function (err, name) {

if(err)

console.log('No-name user');

else

{

entry.get('picture', function (err, pict) {

if(err)

console.log('No picture for user' + name);

else

{

arr.push({name:name, picture:pict});

}

});

}

});

продовження додатку А

});

return arr;

}

function getCurrentDate()

{

return Math.round((new Date()).getTime() / 1000);

}

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Send data to specific client \*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

function sendTo(target, sender, message, callback) {

if(target == 'home')

{

io.sockets.emit('privatemessage', { sender : sender, person : target, message: validator.escape(message) });

callback(2);

return;

}

var clients = io.sockets.clients();

clients.forEach(function(entry) {

entry.get('nickname', function (err, name) {

if(err)

console.log('No-name user');

else

{

if(target == name)

{

entry.emit('privatemessage', {sender: sender, person : sender, message: validator.escape(message)});

callback(1);

return;

}

}

});

});

callback(0);

}

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Starting server \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

server.listen(3700);

console.log('Server started at port 3700');

**ДОДАТОК Б**

**Код клієнтської частини програми**

$( document ).ready(function() {

var messages = [];

var socket = io.connect('http://localhost:3700');

var userlist = document.getElementById("userlist");

var sendButton = document.getElementById("send-message-button");

var addUserButton = document.getElementById("add-user");

var field = document.getElementById("message-input-form");

var chat\_box = document.getElementById("chat-box");

var myname = undefined;

var content = document.getElementById("content");

var name = document.getElementById("name");

var users\_1 = [];

var userhash = $.cookie("userhash");

var pageNum = 1;

var target = 'home';

/\*

\*\*\*\*\*\*\*\*\* Socket.IO handlers \*\*\*\*\*\*\*\*\*

\*/

socket.on('connect', function (){

console.log('successfully established a working connection ');

socket.emit('join', { hash: $.cookie("userhash"), message: 'CONNECTED'});

});

socket.on('userlist', function (data){

users\_1.forEach(function(usr) {

removeUser(usr.name);

}

);

userlist.innerHTML = '';

var arr = data.userlist;

arr.forEach(function(entry) {

users\_1.push(entry);

addUser(entry.name, entry.picture);

}

);

});

socket.on('previousmessages', function (data){

$('#myModalHider').modal('hide');

продовження додатку Б

var arr = data.messages;

for(var i = 0; i < arr.length; i++) {

addDatedMessage(arr[i].target, arr[i].name, arr[i].text, arr[i].date, true);

}

});

socket.on('searchresults', function (data){

var arr = data.messages;

for(var i = 0; i < arr.length; i++) {

addDatedMessage(arr[i].target, arr[i].name, arr[i].text, arr[i].date, true);

}

$('#pageTab a[href="#pmSearch"]').tab('show');

});

socket.on('userlogout', function (data){

removeUser(data.nickname);

});

socket.on('identifymessage', function (data) {

if(data.message && data.name) {

myname = data.name;

addDatedMessage('home', data.sender, data.message);

}

});

socket.on('privatemessage', function (data) {

if(data.message) {

addDatedMessage(data.person, data.sender, data.message, data.date);

addNotificationToTab(data.person);

}

});

/\*

\*\*\*\*\*\*\*\*\* Function KeyPress\*\*\*\*\*\*\*\*\*

\*/

keyPressed = function(e) {

if(e.keyCode == 13) {

sendMessage();

}

};

/\*

\*\*\*\*\*\*\*\*\* On Send button clicked\*\*\*\*\*\*\*\*\*

\*/

sendButton.onclick = function() {

sendMessage();

продовження додатку Б

};

/\*

\*\*\*\*\*\*\*\*\* Send message function\*\*\*\*\*\*\*\*\*

\*/

sendMessage = function() {

if(field.value == "")return;

socket.emit('sendto', { hash: $.cookie("userhash"), message: field.value, target: target});

field.value = "";

}

/\*

\*\*\*\*\*\*\*\*\* Returns current date\*\*\*\*\*\*\*\*\*

\*/

function getDateTime(date) {

var now = (typeof date !== 'undefined')?new Date(date\*1000):new Date();

var year = now.getFullYear();

var months = ['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec'];

var day = now.getDate();

var hour = now.getHours();

var minute = now.getMinutes();

var second = now.getSeconds();

if(day.toString().length == 1) {

var day = '0'+day;

}

if(hour.toString().length == 1) {

var hour = '0'+hour;

}

if(minute.toString().length == 1) {

var minute = '0'+minute;

}

if(second.toString().length == 1) {

var second = '0'+second;

}

var dateTime = '<i class="fa fa-clock-o"></i>' + hour+':'+minute+':'+second;

if (day !== new Date().getDate() && now.getMonth() !== new Date().getMonth())

dateTime = day + ' ' + months[now.getMonth()] + ' on &nbsp;' + dateTime;

return dateTime; }

продовження додатку Б

/\*

\*\*\*\*\*\*\*\*\* Adding new user to user panel\*\*\*\*\*\*\*\*\*

\*/

addUser = function(u\_name,u\_image) {

var userbox = '<li id = "u\_' + u\_name + '">';

userbox += '<img src="' + u\_image+ '" alt="user image" class="userimage"/>';

userbox += '<span class="text">'+u\_name+'</span>';

userbox += '<div class="tools">';

userbox += '<i class="fa fa-edit" onClick=addTab("'+u\_name+'",true);></i>';

userbox += '</div>';

userbox += '</li>';

userlist.innerHTML += userbox;

};

/\*

\*\*\*\*\*\*\*\*\* Load Previous Messages\*\*\*\*\*\*\*\*\*

\*/

loadPreviousMessages = function(uname) {

socket.emit('getpreviousmessages', { hash: $.cookie("userhash"), target: uname});

};

/\*

\*\*\*\*\*\*\*\*\* Adding new private message to chatpanel\*\*\*\*\*\*\*\*\*

\*/

addDatedMessage = function(u\_name, from, message\_text, date, before) {

var uimg = undefined;

uimg = getImage(from);

var msg = '<div class="item">';

msg += ' <img src="'+ uimg +'" alt="user image" class="offline"/>';

msg += '<p class="message">';

msg += '<a href="#" class="name">';

msg += '<small class="text-muted pull-right">';

msg += getDateTime(date);

msg += '</small>';

msg += from;

msg += ' </a>';

msg += message\_text;

msg += '</p>';

продовження додатку Б

msg += '</div>';

addTab(u\_name);

var \_box = document.getElementById('chat-box-'+u\_name);

if(typeof before !== 'undefined' )

\_box.innerHTML = msg + \_box.innerHTML;

else

\_box.innerHTML += msg;

\_box.scrollTop = \_box.scrollHeight;

};

doSearch = function()

{

if( $('#search-query').val() == '')return false;

socket.emit('getsearch', { search: $('#search-query').val()});

$('#search-query').val('');

$('#pageTab li a[href="#pmSearch"] .close').trigger('click');

return false;

};

field.onkeyup = keyPressed;

/\*

\*\*\*\*\*\*\*\*\* Creating new tab\*\*\*\*\*\*\*\*\*

\*/

addTab = function(userid, by\_click)

{

if(myname == userid)return;

if(document.getElementById('chat-box-'+userid))

{

console.log('Tab "chat-box-'+userid+'" now found!')

return;

}

pageNum++;

$('#pageTab').append(

$('<li><a href="#pm'+userid+'" data-toggle="tab">'+userid+'&nbsp;<span class="label label-success"

продовження додатку Б

id="notif-'+userid+'"></span> &nbsp;<button class="close" title="Remove this page" type="button">×</button></a></li>'));

var pagehtml = '<div class="tab-pane fade" id="pm'+userid+'">'+

'<div class="box box-success">'+

'<div class="box-header">'+

'<h3 class="box-title"><i class="fa fa-comments-o"></i>&nbsp;'+userid+'</h3>'+

'<div class="box-tools pull-right" data-toggle="tooltip" title="Status">';

if(userid !== 'Search')

pagehtml += '<button type="button" class="btn btn-default" id="previous"> <span class="glyphicon glyphicon-backward"></span>&nbsp;Previous</button>';

pagehtml += '</div>'+

'</div>'+

'<div class="box-body chat" id="chat-box-'+userid+'">'+

'</div><!-- /.chat -->'+

'</div>'+

'</div>';

$('#pageTabContent').append( pagehtml );

if(typeof by\_click !== 'undefined' )

{

$('#pageTab a[href="#pm'+userid+'"]').tab('show');

target = userid;

}

}

}