

**МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ**

**ОТЧЕТ
по лабораторной работе №4
по дисциплине «Web-технологии»
Тема: Модуль приложения «Участие в аукционе картин»**

Студент гр.8303
Преподаватель

Рудько Д.Ю.
Беляев С.А.

Санкт-Петербург
2020

Цель работы.

Изучить возможности применения jQuery UI для создания интерфейса пользователя, обработку ошибок, ведение журналов ошибок, реализацию взаимодействия приложений с использованием web-сокетов.

Задание.

Для достижения поставленной цели требуется решить следующие задачи:

- 1) Разработка интерфейса web-приложения.
- 2) Создание web-сервера на основе express, настройка маршрутов, подготовка и обработка REST-запросов (серверная часть).
- 3) Создание шаблонов web-страниц, указание путей подключения js-файлов.
- 4) Разработка клиентских js-файлов с использованием библиотек jQuery и jQuery UI.
- 5) Создание web-сокета для отправки сообщений всем клиентам.

Ход выполнения.

Создан web-сервер на основе express. Созданы шаблоны страницы входа, а так же страниц пользователей и администратора, реализован ввод из json файлов.

На сервере обработаны все необходимые запросы. Создан веб-сокет для отправки сообщений клиентам. Клиенты подключены к сокету.

Страница входа

Welcome to Аукцион

Имя пользователя

Войти

Страница администратора (до начала торгов)

Вы зашли как admin

Рудько

Счет: 10000000

Капустин

Счет: 1000

Звёздная ночь



Владелец:

Цена: 1000000

9 вал



Владелец:

Цена: 1111

Бурлаки на Волге



Владелец:

Цена: 100000

Рудько

83




Рудько-присоединился(лась)

Когда аукцион начнется вас перебросит на страницу торгов, где после изучения товара вы сможете сделать ставки

Удачных ставок!

Страница администратора во время торгов

Вы зашли как admin

Рудько	Капустин	Звёздная ночь	9 вал	Бурлаки на Волге	Аукцион начался
Счет: 8898487	Счет: 1000				Рудько поднял цену на картину Звёздная ночь, новая цена 1000013
		Владелец: Рудько Цена: 1000039	Владелец: Рудько Цена: 1444	Владелец: Рудько Цена: 100030	Рудько поднял цену на картину Звёздная ночь, новая цена 1000026
					Рудько поднял цену на картину Звёздная ночь, новая цена 1000039
					Рудько поднял цену на картину 9 вал, новая цена 1222
					Рудько поднял цену на картину 9 вал, новая цена 1333
					Рудько поднял цену на картину 9 вал, новая цена 1444
					Рудько поднял цену на картину Бурлаки на Волге, новая цена 100010
					Рудько поднял цену на картину Бурлаки на Волге, новая цена 100020

Страница участника в начале торгов

Аукцион:

Текущей счет: 10000000

Звёздная ночь	9 вал	Бурлаки на Волге
		
Владелец: - Цена: 1000000 Min / Max: 13 / 100	Владелец: - Цена: 1111 Min / Max: 111 / 1111	Владелец: - Цена: 100000 Min / Max: 10 / 100
<div>13</div> <div>Ставка</div>	<div>111</div> <div>Ставка</div>	<div>10</div> <div>Ставка</div>

Страница со списком участников

Рудько

Счет: 8898487

Картины: Звёздная
ночь 9 вал Бурлаки
на Волге

Капустин

Счет: 1000

Картины:

Выводы.

В ходе выполнения лабораторной работы было написано приложение для аукциона картин, изучены возможности jQuery UI, использование web-сокетов.

ПРИЛОЖЕНИЕ А. ИСХОДНЫЙ КОД

app.js

```
var createError = require('http-errors');
var express = require('express');
var path = require('path');
var cookieParser = require('cookie-parser');
var logger = require('morgan');
var http = require('http');

var indexRouter = require('./routes/router');

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

app.set('views', path.join(__dirname, 'views'));
app.set('view engine', 'pug');

app.use(logger('dev'));
app.use(express.json());
app.use(express.urlencoded({ extended: false }));
app.use(cookieParser());
app.use(express.static(path.join(__dirname, 'public')));

app.use('/', indexRouter);

// catch 404 and forward to error handler
app.use(function(req, res, next) {
  next(createError(404));
});

// error handler
app.use(function(err, req, res, next) {
  // set locals, only providing error in development
  res.locals.message = err.message;
  res.locals.error = req.app.get('env') === 'development' ? err : {};

  // render the error page
  res.status(err.status || 500);
  res.render('error');
});

server.listen(3000, function() {
  console.log('Started, port: 3000')
})

var players = require('./public/json/players.json');
var arts = require('./public/json/arts.json');
var setting = require('./public/json/setting.json');
var count = 0;
var now = '';

io.sockets.on('connection', function (socket){
  // общая рассылка приветствие
  socket.on('echo', (data) => {
    user = data.name;
    const time = setting.myAuc.timeBegin;
    const Minute = time.split(':')[1] *1
    socket.broadcast.json.emit('welcome', { message: `${data.name}-присоединился(лась)`, setting:setting, arts: arts, users:players, start:Minute});
    socket.json.emit('welcome', { message: `${data.name}-присоединился(лась)`, setting:setting, arts: arts, users:players, start:Minute});
  });
  // начало аукциона
  socket.on('startAuction', () => {
    const time = setting.myAuc.timeBegin;
    const Minute = time.split(':')[1] *1;
    const Hour = time.split(':')[0] *1;
    const nowDate = new Date();
    const nowHour = nowDate.getHours() *1;
```

```

const nowMinute = nowDate.getMinutes() *1;
if(Minute === nowMinute && Hour === nowHour){
    now = "<p>" + 'Аукцион начался';
    socket.broadcast.emit('start', {start: true, all: players.arr});
    socket.emit('start', {start: true, all: players.arr});
}
else{
    socket.emit('start', {start: false});
}
});
// сделать ставку
socket.on('bid', (data) => {
    let newPrice = data.upPrice *1;
    let oldPrice;
    let wid = "-";
    let flag = 0;
    // получили текущей счет usera
    let cash = 0;
    let r = 0;
    let id = 0;
    for(let i = 0; i < players.arr.length; i++){
        if(data.playerr === players.arr[i].name){
            cash = players.arr[i].money;
            r = i;
        }
    }
    cash *= 1;
    if(newPrice > cash){
        flag = -1;
        socket.broadcast.emit('messageBid', {flag:false});
        socket.emit('messageBid', {flag:false});
    }
    else {
        // находим старую цену
        for(let i = 0; i < arts.arr.length; i++){
            if(arts.arr[i].name === data.name){
                oldPrice = arts.arr[i].current_price;
            }
        }
        oldPrice *= 1;
        if(oldPrice >= newPrice){
            flag = -1;
            socket.broadcast.emit('messageBid', {flag:false});
            socket.emit('messageBid', {flag:false});
        }
        else {
            wid = data.playerr;
            for(let g = 0; g < players.arr.length; g++){
                if(players.arr[g].name === data.picUser){
                    let curmo = players.arr[g].money;
                    curmo *= 1;
                    curmo += oldPrice *1;
                    players.arr[g].money = curmo;
                }
            }
            // списываем у нового сумму
            cash -= newPrice;
            players.arr[r].money -= newPrice;
            // отдает клиенту
            for(let i = 0; i < arts.arr.length; i++){
                if(arts.arr[i].name === data.name){
                    arts.arr[i].current_price = newPrice;
                    arts.arr[i].owner = wid;
                    console.log(arts.arr[i].name + " " + arts.arr[i].owner);
                    id = arts.arr[i].id;
                }
            }
            now = "<p>" + wid + ' поднял цену на картину ' + data.name + ',
новая цена ' + newPrice;
            socket.broadcast.emit('messageBid', {flag:true, price:newPrice,
picUser:wid, cash:players.arr, ID:id});

```

```

        socket.emit('messageBid', {flag:true, price:newPrice,
picUser:wid, cash:players.arr, ID:id});
    }
}
});
// проверка времени прихода usera
socket.on('UserInAuction', (data) => {
    const dateSettings = setting.myAuc.dateBegin;
    const timeSettings = setting.myAuc.timeBegin;
    const settingsMinute = timeSettings.split(':')[1] *1;
    const settingsHour = timeSettings.split(':')[0] *1;
    const settingsDay = dateSettings.split('-')[2] *1;
    const settingsMonth = dateSettings.split('-')[1] *1;
    const settingsYear = dateSettings.split('-')[0] *1;
    let flag = 0;

    if(data.userYear === settingsYear && data.userMonth === settingsMonth){
        if(data.userDay < settingsDay){flag = -1;}
        if(data.userDay > settingsDay){flag = 1;}
        if(data.userDay === settingsDay){
            if(data.userHour > settingsHour){flag = 1;}
            if(settingsHour - data.userHour > 1){flag = -1;}
            if(settingsHour - data.userHour === 1 || settingsHour -
data.userHour === 0){
                let defferenceMinut = data.userMinute - settingsMinute;
                if(defferenceMinut < 0){
                    defferenceMinut *= -1;
                    if(defferenceMinut > 3){flag = -1;}
                }
                if(defferenceMinut > 3){flag = 1;}
            }
        }
    }
    else {
        let defferernceYear = data.userYear - settingsYear;
        if(defferernceYear < 0){flag = -1;}
        if(defferernceYear > 0){flag = 1;}
    }

    if(flag !== 0){socket.emit('message', {flag:flag});}
    if(flag === 0){socket.emit('start',{start: false});}

});
socket.on('AfterAuction', ()=>{
    count++;
    if(count === 1) {now = "<p>" + 'Аукцион окончен';}
    socket.emit('End', {count: count});
});

socket.on('toAdmin', (data)=> {
    const DN = now;
    now = '';
    socket.emit('admin2310', {pics:arts.arr, players:players.arr, now:DN});
});
});

```

module.exports = **app**;

router.js

```

var express = require('express');
var router = express.Router();

var players = require('../public/json/players.json');
var arts = require('../public/json/arts.json');

```

```

/* GET home page. */
router.get('/', function(req, res) {

```

```

    res.render('index', { title: 'Аукцион' });
});

var cname;
router.post('/in', function (req, res){
    let name = req.body.name;
    if(name === "admin"){
        res.redirect('administration');
        return;
    }
    else {
        for(let i = 0; i < players.arr.length; i++){
            if(req.body.name*1 === players.arr[i].id*1){
                cname = players.arr[i].name;
                res.redirect('main');
                return;
            }
        }
    }
    res.end("NO");
});

router.get('/main', function (req, res){
    res.render('main', {title: cname});
})

router.get('/players/:count' , function (req, res){
    let count = req.params.count *1;
    if(count === 1) {
        for (let i = 0; i < arts.arr.length; i++) {
            if (arts.arr[i].owner !== "") {
                for (let j = 0; j < players.arr.length; j++) {
                    if (arts.arr[i].owner === players.arr[j].name) {
                        players.arr[j].auction += (arts.arr[i].name + '\n');
                    }
                }
            }
        }
    }
    res.render('players',{players: players.arr});
});

router.get('/auction/:user', function (req, res){
    let user = req.params.user;
    console.log(user);
    res.render('auction', {arts: arts.arr, user: user, players: players.arr});
})

router.get('/administration', function (req,res){
    res.render('administration', {title: 'admin', arts: arts.arr,
    players:players.arr});
});

module.exports = router;

```

socket.js

```

var socket = io.connect('http://localhost:3000');

socket.on('connect',function(){
    var text = $('title').text();
    if(document.location.href === "http://localhost:3000/main" ||
    document.location.href === "http://localhost:3000/administration"){
        document.cookie = `username=${text}`
    }
    socket.emit('echo', {name:text});
});

socket.on('connect',function(){
    if(document.location.href === "http://localhost:3000/main") {
        let data = new Date();
        const userDay = data.getDate() * 1;
    }

```



```

const userMonth = data.getMonth() * 1 + 1;
const userYear = data.getFullYear() * 1;
const userHour = data.getHours() * 1;
const userMinute = data.getMinutes() * 1;
socket.emit('UserInAuction', {
    userDay: userDay,
    userMonth: userMonth,
    userYear: userYear,
    userHour: userHour,
    userMinute: userMinute
});
});

socket.on('welcome', function(data) {
    var text = "<p>" + data.message;
    $('#msg_field').append(text);
    $('#start').html(data.start);
});

socket.on('start', (data)=>{

    const userName = document.cookie.split('=')[1];
    const r = document.location.href.split('/')[3];
    if (data.start === true && userName !== 'admin' && r !== 'auction') {
        document.location.href = `/auction/${userName}`;
    } else {
        socket.emit('startAuction');
    }
});

socket.on('End', (data) => {
    document.location.href = `/players/${data.count}`;
    socket.broadcast.disconnect();
    socket.disconnect();
});

socket.on('message', (data) => {
    if(data.flag *1 === 1){alert("Вы опаздали на аукцион");}
    if(data.flag *1 === -1){alert("Вы пришли слишком рано");}
    document.location.href = "/";
});

socket.on('messageBid', (data) =>{
    const userName = document.cookie.split('=')[1];
    if(data.flag === true){
        console.log("IN MESSAGEBID");
        console.log(data);
        //меняем цену
        $('#price${data.ID}`).html(`Цена: ${data.price}`);
        // меняем владельца
        $('#widower${data.ID}`).html(`Владелец: ${data.picUser}`);
        // меняем тек. счет
        const pl1 = document.cookie.split('=')[1];
        for(let i = 0; i < data.cash.length; i++){
            if(data.cash[i].name === pl1){
                $('#cash').html(`Текущей счет: ${data.cash[i].money}`);
            }
        }
    }
});

function Bid(button){
    const ID = button.id.split(" ")[1];
    const name = $('#PicName${ID}`).html().split('>')[2];
    const upPrice = $('#UpPrice${ID}`).val() *1;
    const oldPrice = $('#price${ID}`).html().split(' ')[1] *1;
    const newPrice = oldPrice + upPrice;
    const picUser = $('#widower${ID}`).html().split(' ')[1];
    const pl = document.cookie.split('=')[1];
    const up = $('#minMax${ID}`).html().split(' ')[5] *1;
    const down = $('#minMax${ID}`).html().split(' ')[3] *1;

```

```

        if(upPrice >= down && upPrice <= up){
            socket.emit('bid', {name: name, playerr: pl, picUser:picUser, upPrice:
newPrice});
        }
    }
}

```

adminSocket.js

```

var soc = io.connect('http://localhost:3000');

soc.on('connect', ()=>{
    soc.emit('toAdmin', {countD: 'fromAdmin2310'});
});

soc.on('admin2310', (data)=>{

    var count = 0;

    for(let j = 0; j < data.pics.length; j++){
        if(data.pics[j].inAuction === true) {
            count++;
            $('#AdminPrice${data.pics[j].id}`).html(`Цена: $
{data.pics[j].current_price}`);
            $('#AdminWidower${data.pics[j].id}`).html(`Владелец: $
{data.pics[j].owner}`);
        }
    }

    for(let i = 0; i < data.players.length; i++){
        $('#cash${data.players[i].id}`).html(`Счет: ${data.players[i].money}`);
    }

    if(data.now !== '') {
        $('#new').append(data.now);
        count = '';
    }

    soc.emit('toAdmin', {countD: count});
});

```

main.js

```

setTimeout(timer, 10);

function timer(){
    const d = new Date();
    const s = $('#start').html() *1;
    const r = (s - d.getMinutes())*60 - d.getSeconds();
    if(r > 0)
        $('#timer').html(r);
    setTimeout(timer, 1000);
}

```

auction.js

```

$(document).ready(()=> {
    $( ".column" ).sortable({
        connectWith: ".column",
        handle: ".portlet-header",
        cancel: ".portlet-toggle",
        placeholder: "portlet-placeholder ui-corner-all"
    });

    $( ".portlet" )
        .addClass( "ui-widget ui-widget-content ui-helper-clearfix ui-corner-
all" )
        .find( ".portlet-header" )
        .addClass( "ui-widget-header ui-corner-all" )
        .prepend( "<span class='ui-icon ui-icon-circle-triangle-s portlet-
toggle'></span>" );

    $( ".portlet-toggle" ).on( "click", function() {
        var icon = this;
        icon.toggleClass( "ui-icon-circle-triangle-s ui-icon-circle-triangle-n"

```

```

);
    icon.closest( ".portlet" ).find( ".portlet-content" ).toggle();
});

$('#progressbar').progressbar({value: 0});
let flag = true;
function progress(){
    $("#progressbar > div").css({ 'background': 'Green' });
    var val = $('#progressbar').progressbar("value") || 0;
    $('#progressbar').progressbar("value", val+30);
    if(val < 95)
        setTimeout(progress, 1000);
    else {
        $('#l1').html("Аукцион: ");
        $('.BUpPrice').css({'pointer-events' : 'visible'});
        $('#progressbar').progressbar("value", 0);
        setTimeout(progressTwo, 1000);
    }
}

$('.BUpPrice').css({'pointer-events' : 'none'});
$('.disable').css({'pointer-events' : 'none'});

setTimeout(progress, 1000);

function progressTwo(){

    $("#progressbar > div").css({ 'background': 'Green' });
    let val = $('#progressbar').progressbar("value") || 0;
    $('#progressbar').progressbar("value", val+2);
    if(val < 100) {
        setTimeout(progressTwo, 1000);
    }
    else {
        $('.disable').css({'pointer-events': 'initial'});
        alert("Аукцион окончен, спасибо за участие");
        socket.emit('AfterAuction');
    }
}

});

```

players.js

```

$(document).ready(()=> {
    $( ".column" ).sortable({
        connectWith: ".column",
        handle: ".portlet-header",
        cancel: ".portlet-toggle",
        placeholder: "portlet-placeholder ui-corner-all"
    });

    $( ".portlet" )
        .addClass( "ui-widget ui-widget-content ui-helper-clearfix ui-corner-
all" )
        .find( ".portlet-header" )
        .addClass( "ui-widget-header ui-corner-all" )
        .prepend( "<span class='ui-icon ui-icon-circle-triangle-s portlet-
toggle'></span>" );

    $( ".portlet-toggle" ).on( "click", function() {
        var icon = $(this);
        icon.toggleClass( "ui-icon-circle-triangle-s ui-icon-circle-triangle-n"
    );
        icon.closest( ".portlet" ).find( ".portlet-content" ).toggle();
    });
});

```

administration.pug

extends layout

```
block content
  script(src="/javascripts/adminSocket.js")
  div
    p Вы зашли как #{title}
    for user in players
      div.column
        div.portlet
          div.portlet-header #{user.name}
          div.portlet-content
            p(id="cash" + user.id) Счет: #{user.money}
    p
    for art in arts
      if(art.inAuction === true)
        div.column
          div.portlet
            div(id="AdminPicName" + art.id ).portlet-header #{art.name}
            div.portlet-content
              div(class="right-panel-wrapper")
                a(href=art.pt target="_blank")
                  img.image(src=art.pt class="indBook"
href=art.pt)
                  p(id="AdminWidower" + art.id) Владелец: -
                  p(id='AdminPrice' + art.id) Цена: #{art.beginning_price}
            div( class='Current' id="new")
```

auction.pug

extends layout

```
block content
  script(src='/javascripts/auction.js')
  script(src="/javascripts/socket.js")
  lable(id='l1') Изучение товара:
  div(id="progressbar").ui-progressbar
  p
  for pl in players
    if(pl.name === user)
      div(id="cash") Текущей счет: #{pl.money}
  for art in arts
    if(art.inAuction === true)
      div.column
        div.portlet
          div(id="PicName"+art.id ).portlet-header #{art.name}
          div.portlet-content
            div(class="right-panel-wrapper")
              a(href=art.pt target="_blank")
                img.image(src=art.pt class="indBook"
href=art.pt)
                p(id="widower"+art.id) Владелец: -
                p(id='price'+art.id) Цена: #{art.beginning_price}
                p(id='minMax' + art.id) Min / Max: #{art.min_step} /
#{art.max_step}
                div
                  input(class='UpPrice' type="number"
id="UpPrice"+art.id min=art.min_step value=art.min_step)
                  button(class='BUpPrice' id="BUpPrice "+art.id
onclick="Bid(this)") Ставка
```

error.pug

extends layout

```
block content
  h1= message
  h2= error.status
  pre #{error.stack}
```

header.pug

```
div(class="header")
  nav(class="menu")
    ul(class="hr")
      li: a(href='/main' class='disable' id="dfg") Главная
      li: a(href='/auction' class='disable') Аукцион
      li: a(href='/players' class='disable') Участники
```

index.pug

```
block content
  link(rel="icon" href="/images/favicon.ico" type='image/x-icon')
  link(rel="shortcut icon" href="/images/favicon.ico" type='image/x-icon')
  div
    div
      p Welcome to #{title}
    div(id='input')
      form(action="/in" method="POST")
        p
          label Имя пользователя
        p
          input(name = "name" required)
        p
          button(type="submit") Войти
```

layout.pug

```
doctype html
html
  head
    title= title
    style
      include ../node_modules/jquery-ui-dist/jquery-ui.css
    script(src="/socket.io/socket.io.js")
    script(src = "https://code.jquery.com/jquery-1.12.4.js")
    script(src="https://code.jquery.com/ui/1.12.1/jquery-ui.js")
    link(rel="stylesheet" href="/stylesheets/style.css")
    link(rel="icon" href="/images/favicon.ico" type='image/x-icon')
    link(rel="shortcut icon" href="/images/favicon.ico" type='image/x-icon')
  body
    block content
```

main.pug

extends layout

```
block content
  script(src="/javascripts/socket.js")
  script(src="/javascripts/main.js")
  h1 #{title}
  hr
  div(id="start")
  div(id="timer")
  hr
  div(id="msg_field")
  hr
  div
    p Когда аукцион начнется вас перебросит на страницу торгов, где после изучения товара вы сможете сделать ставки
    p Удачных ставок!
```

players.pug

extends layout

```
block content
  script(src="/javascripts/socket.js")
  script(src="/javascripts/players.js")
  for user in players
    div.column
      div.portlet
        div.portlet-header #{user.name}
        div.portlet-content
          p Счет: #{user.money}
```

р Картины: #{user.auction}