

```

<?php
/**
 * ТДС корзины с объединением всех шагов на одной странице v. 6.2
 * обязательно использовать с соответствующей страницей настроек ТДС
 * для вывода типов доставки и пересчета используется ajax
 *
 * @author Kuts Artem, KAD Systems (©) 2014
 * @date 01-09-2014
 */

// Все настройки переехали во вкладку Настройки

/*
 * Для использования captcha необходимо в xsl шаблон вывода платежных систем
 * вставить код:

        <!-- Обработка CAPTCHA -->
            <xsl:if test="captcha_id != 0 and siteuser_id = 0">
                <div class="row">
                    <div class="caption"></div>
                    <div class="field">
                        
                        <div class="captcha">
                             <span
onclick="$('#comment').updateCaptcha('{captcha_id}', 30); return
false">Показать другое число</span>
                        </div>
                    </div>
                </div>
                <div class="row">
                    <div class="caption">
Контрольное число<sup><font color="red">*</font></sup>
                    </div>
                    <div class="field">
                        <input type="hidden" name="captcha_id"
value="{captcha_id}"/>
                        <input type="text" name="captcha" size="15"/>
                    </div>
                </div>
            </xsl:if>

*/

?>

<style>
input.input_error{
    border: 1px solid red!important;
}
</style>

```

```

<script>
    $(document).ready(function(){
        // Получить данные формы
        getFormData = function($type)
        {
            return $("form[name='order']").serialize() + "&ajaxget=" +
$type;
        }
        // Загрузить корзину
        refreshCart = function()
        {
            $.loadingScreen('show');
            var formData = getFormData("");
            $("#cart").load(
                "",
                formData + "&ajaxget=cart"
            ,
            function(){
                $.loadingScreen('hide');
                <?php
                if (SHORTCART_CARTMODE == 2)
                {
                    ?>
                        refreshDeliveries();
                    <?php
                }
                ?>
            }
        );

        // Наблюдатель после перезагрузки корзины
        if (typeof afterReloadCart == 'function')
        {
            afterReloadCart();
        }
    }
    // Загрузить способы оплаты
    refreshPayments = function()
    {
        // Запоминаем выбранный
        var $active_val = $("#payment
input[type=radio]:checked").data("id");
        $.loadingScreen('show');
        var formData = getFormData("payment");
        $("#payment").load(
            "",
            formData
        ,
        function(){
            // Возвращаем выбранный
            var $radiobtn = $("#payment input[type='radio']")[data-

```

```

id='" + $active_val + "']");
        if ($radiobtn != "undefined")
        {
            $radiobtn.attr("checked", "checked");
        }
        $.loadingScreen('hide');
    }
    );
}
// Загрузить способы доставки
refreshDeliveries = function()
{
    // Запоминаем выбранный
    var $active_val = $("#delivery
input[type=radio]:checked").data("id");
    $.loadingScreen('show');
    var formData = getFormData("delivery");
    $("#delivery").load(
        "",
        formData
    ,
        function(){
            $.loadingScreen('hide');
            // Возвращаем выбранный
            var $radiobtn = $("#delivery input[type='radio'][data-
id='" + $active_val + "']");
            if ($radiobtn != "undefined")
            {
                $radiobtn.attr("checked", "checked");
            }
            refreshPayments();
        }
    );
}
$("#[name^='quantity'], [name='coupon_text']").live('change',
function(){
    refreshCart();
});
// Пересчитать
$('#[name="recount"]').live('click', function(){
    refreshCart();
    return false;
});
// Подгрузка типов доставки
$("#shop_country_location_city_id, #shop_country_location_id,
#shop_country_id").change(function(){
    refreshDeliveries();
});
// Подгрузка платежных систем
$("#[name='shop_delivery_condition_id']").live("click", function(){
    refreshPayments();
});

```

```

    });
    // Обновление районов
    $.loadLocations('.', $("#shop_country_id").val());
    $("#shop_country_id").change();

    // Валидация
    //Если не работает валидация, закомментировать строчку ниже
    if (typeof $.validate == 'function')
    {
        $("form[name='order']").validate({
            focusInvalid: true,
            errorClass: "input_error",
            errorPlacement: function(error, element) {}
        })
    }
    });
</script>

<?php
$Shop_Cart_Controller_Show = Core_Page::instance()->object;

$Shop = $Shop_Cart_Controller_Show->getEntity();

// -----
// Подготовка редиректа для PayPal
// -----
if (isset($_REQUEST['paymentType']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getRequest('order_id'));

    $Shop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (!is_null($Shop_Order->id))
    {
        // Вызов обработчика платежной системы
        Shop_Payment_System_Handler::factory($Shop_Order->Shop_Payment_System)
            ->shopOrder($Shop_Order)
            ->paymentProcessing();
    }
}

// -----
// Подготовка редиректа для Оплаты с лицевого счета
// -----
if (isset($_REQUEST['Pay']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getRequest('order_id'));

    $Shop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

```

```

        if (!is_null($oShop_Order->id))
        {
            // Вызов обработчика платежной системы
Shop_Payment_System_Handler::factory($oShop_Order->Shop_Payment_System)
                ->shopOrder($oShop_Order)
                ->paymentProcessing();
        }
    }

// -----
// Обработка уведомления об оплате от RBK Money
// -----
if ((isset($_POST['paymentStatus']) ||
isset($_GET['PayPalOrderConfirmation'])) &&
!isset($_REQUEST['paymentType']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getPost('orderId'));

    $oShop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (!is_null($oShop_Order->id))
    {
        // Вызов обработчика платежной системы
Shop_Payment_System_Handler::factory($oShop_Order->Shop_Payment_System)
                ->shopOrder($oShop_Order)
                ->paymentProcessing();
    }
}

// -----
// Обработка уведомления об оплате от WebMoney
// -----
if (isset($_POST['LMI_PAYEE_PURSE']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getPost('LMI_PAYMENT_NO'));

    $oShop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (!is_null($oShop_Order->id))
    {
        // Вызов обработчика платежной системы
Shop_Payment_System_Handler::factory($oShop_Order->Shop_Payment_System)
                ->shopOrder($oShop_Order)
                ->paymentProcessing();
    }
}

// -----
// Обработка уведомления об оплате от ROBOKASSA

```

```

// -----
if (isset($_REQUEST['SignatureValue']) && isset($_REQUEST['Culture']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getRequest('InvId'));

    $oShop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (!is_null($oShop_Order->id))
    {
        // Вызов обработчика платежной системы
        Shop_Payment_System_Handler::factory($oShop_Order->Shop_Payment_System)
            ->shopOrder($oShop_Order)
            ->paymentProcessing();
    }
}
// -----

// -----
// Обработка уведомления об оплате от Authorize.net
// -----
if (isset($_POST['x_response_code']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getPost('order_id'));

    $oShop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (!is_null($oShop_Order->id))
    {
        // Вызов обработчика платежной системы
        Shop_Payment_System_Handler::factory($oShop_Order->Shop_Payment_System)
            ->shopOrder($oShop_Order)
            ->paymentProcessing();
    }
}

// -----
// Обработка формы "Оплата через систему QIWI Кошелек"
// -----
if (isset($_POST['qiwi_payment_options']))
{
    // Получаем ID заказа
    $order_id = intval(Core_Array::getPost('order_id'));

    $oShop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (!is_null($oShop_Order->id))
    {
        // Вызов обработчика платежной системы
        Shop_Payment_System_Handler::factory($oShop_Order->Shop_Payment_System)

```

```

        ->shopOrder($oShop_Order)
        ->paymentProcessing();
    return TRUE;
    }
}

// -----
// Вывод информации о статусе платежа после его совершения и перенаправления с
// платежной системы
// -----
if (isset($_REQUEST['payment']) || (isset($_GET['action']) &&
($_GET['action'] == 'PaymentSuccess' || $_GET['action'] == 'PaymentFail'))))
{
    // Получаем ID заказа
    if (isset($_REQUEST['order_id']))
    {
        $order_id = intval(Core_Array::getRequest('order_id'));
    }
    //от Яндекса
    elseif(isset($_GET['orderNumber']))
    {
        $order_id = intval(Core_Array::getGet('orderNumber'));
    }
    else
    {
        $order_id = intval(Core_Array::getRequest('InvId'));
    }

    $oShop_Order = Core_Entity::factory('Shop_Order')->find($order_id);

    if (Core::moduleIsActive('siteuser') && !SHORTCART_IGNOREUSERS)
    {
        $siteuser_id = 0;

        $oSiteuser = Core_Entity::factory('Siteuser')->getCurrent();
        if ($oSiteuser)
        {
            $siteuser_id = $oSiteuser->id;
        }
    }
    else
    {
        $siteuser_id = FALSE;
    }

    // Если заказ принадлежит текущему авторизованному пользователю
    if ($oShop_Order->siteuser_id == $siteuser_id)
    {
        if (Core_Array::getRequest('payment') == 'success' ||
Core_Array::getRequest('action') == 'PaymentSuccess')
        {

```

```

        ?><h1>Подтверждение платежа</h1>
        <p>Спасибо, информация об оплате заказа <strong>№ <?php echo
$Shop_Order->invoice?></strong>
получена.</p>
        <?php
    }
    else
    {
        ?><h1>Платеж не получен</h1>
        <p>К сожалению при оплате заказа <strong>№ <?php echo
$Shop_Order->invoice?></strong> произошла ошибка.</p>
        <?php
    }
}
// Для случаев, когда отключен модуль "Пользователи сайта"
elseif ($siteuser_id === FALSE)
{
    ?><h1>Подтверждение платежа</h1>
    <p>Благодарим за посещение нашего магазина!</p>
    <?php
}
else
{
    ?><h1>Ошибка</h1>
    <p>Неверный номер заказа!</p>
    <?php
}

// Прерываем выполнение типовой динамической страницы
return TRUE;
}

$xmlName = Core_Array::get(Core_Page::instance()->libParams, 'cartXml');

// Пересчитываем количество товара в корзине
$countGoods = 0;
foreach (Shop_Cart_Controller::instance()->getAll($Shop) as $Good)
{
    $Shop_Item =
Core_Entity::factory('Shop_Item')->find($Good->shop_item_id);
    if(!is_null($Shop_Item->id))
    {
        $countGoods += $Good->quantity;
    }
}

// Если режим в 2 шага, выводим все на главной
$error_message = "";
if (SHORTCART_CARTMODE == 2 && !Core_Array::getPost('step') && $countGoods >
0 || Core_Array::getPost('recount'))
{

```



```

        $step = 1;
    } else
    {
        if (!SHORTCART_USECAPTCHA || (SHORTCART_USECAPTCHA &&
Core_Captcha::valid(Core_Array::getPost('captcha_id'),
Core_Array::getPost('captcha'))))
        {
            $step = Core_Array::getPost('step');
        } else
        {
            $step = 1;
            $error_message = SHORTCART_ERROR_WC;
        }
    }

switch ($step)
{
    // Адрес доставки
    case 1:
        // Все шаги должны содержать единую форму, поэтому из xsl шаблонов формы
желательно убрать
        ?>
            <form name="order" method="POST" action=".">
        <?php
        if (SHORTCART_CARTMODE == 2)
        {
            ?>
                <div id="cart">
            <?php
            $Shop_Cart_Controller_Show
                ->couponText(Core_Array::get(Core_Array::get($_SESSION,
'hostcmsOrder', array()), 'coupon_text'))
                ->xsl(
                    Core_Entity::factory('Xsl')->getByName($xslName)
                )
                ->show();
            ?>
                </div>
            <?php
        }
        // Сбрасываем информацию о последнем заказе
        $_SESSION['last_order_id'] = 0;

        // Адрес доставки
        $Shop_Address_Controller_Show = new
Shop_Address_Controller_Show($oShop);
        $Shop_Address_Controller_Show->xsl(
            Core_Entity::factory('Xsl')->getByName(
                Core_Array::get(Core_Page::instance()->libParams,
'deliveryAddressXsl')
            )

```

```

        )
        ->show();
// /Адрес доставки

// Типы доставки / ajax обработчик в настройках
?>
    <div id="delivery">
    <?php
        $Shop_Delivery_Controller_Show = new
Shop_Delivery_Controller_Show($oShop);
        $Shop_Delivery_Controller_Show
            ->couponText(Core_Array::get(Core_Array::get($_SESSION,
'hostcmsOrder', array()), 'coupon_text'))
            ->setUp()
            ->xsl(
                Core_Entity::factory('Xsl')->getByName(
                    Core_Array::get(Core_Page::instance()->libParams,
'deliveryXsl')
                )
            )
            ->show();
    ?>
    </div>
    <?php
// /Типы доставки
// Платежные системы / ajax обработчик в настройках
    ?>
        <div id="payment">
        <?php
            $Shop_Payment_System_Controller_Show = new
Shop_Payment_System_Controller_Show($oShop);
            if (SHORTCART_USECAPTCHA)
            {
                $Shop_Payment_System_Controller_Show->addEntity(
                    Core::factory('Core_Xml_Entity')
                        ->name('siteuser_id')
                        ->value((Core::moduleIsActive('siteuser') &&
!is_null(Core_Entity::factory('Siteuser')->getCurrent()))?Core_Entity::facto
ry('Siteuser')->getCurrent()->id:0)
                )->addEntity(
                    Core::factory('Core_Xml_Entity')
                        ->name('captcha_id')
                        ->value(Core_Captcha::getCaptchaId())
                );
            }
            $Shop_Payment_System_Controller_Show->addEntity(
                Core::factory('Core_Xml_Entity')
                    ->name('message')
                    ->value($error_message)
            );
        }
    ?>

```

```

        $Shop_Payment_System_Controller_Show
            ->xsl(
                Core_Entity::factory('Xsl')->getByName(
                    Core_Array::get(Core_Page::instance()->libParams,
'paymentSystemXsl')
                )
            )
            ->show();
        ?>
    </div>
<?php
// Платежные системы
?>
    </form>
<?php
break;
// Окончание оформления заказа
case 4:
    $_SESSION['hostcmsOrder']['postcode'] =
strval(Core_Array::getPost('postcode'));
    $_SESSION['hostcmsOrder']['address'] =
strval(Core_Array::getPost('address'));
    $_SESSION['hostcmsOrder']['surname'] =
strval(Core_Array::getPost('surname'));
    $_SESSION['hostcmsOrder']['name'] =
strval(Core_Array::getPost('name'));
    $_SESSION['hostcmsOrder']['patronymic'] =
strval(Core_Array::getPost('patronymic'));
    $_SESSION['hostcmsOrder']['company'] =
strval(Core_Array::getPost('company'));
    $_SESSION['hostcmsOrder']['phone'] =
strval(Core_Array::getPost('phone'));
    $_SESSION['hostcmsOrder']['fax'] =
strval(Core_Array::getPost('fax'));
    $_SESSION['hostcmsOrder']['email'] =
strval(Core_Array::getPost('email'));
    $_SESSION['hostcmsOrder']['description'] =
strval(Core_Array::getPost('description'));
    $_SESSION['hostcmsOrder']['shop_delivery_condition_id'] =
intval(Core_Array::getPost('shop_delivery_condition_id', 0));
    $shop_payment_system_id =
$_SESSION['hostcmsOrder']['shop_payment_system_id'] =
intval(Core_Array::getPost('shop_payment_system_id', 0));
    // Additional order properties
    $_SESSION['hostcmsOrder']['properties'] = array();

    $oShop_Order_Property_List =
Core_Entity::factory('Shop_Order_Property_List', $oShop->id);

    $aProperties = $oShop_Order_Property_List->Properties->findAll();
    foreach ($aProperties as $oProperty)

```

```

        {
            // Св-во может иметь несколько значений
            $aPropertiesValue = Core_Array::getPost('property_' .
            $oProperty->id);

            if (!is_null($aPropertiesValue))
            {
                !is_array($aPropertiesValue) && $aPropertiesValue =
array($aPropertiesValue);
                foreach ($aPropertiesValue as $sPropertyValue)
                {
                    $_SESSION['hostcmsOrder']['properties'][] =
array($oProperty->id, $sPropertyValue);
                }
            }

            // Если выбрана платежная система
            if ($_SESSION['hostcmsOrder']['shop_payment_system_id'])
            {
                Shop_Payment_System_Handler::factory(
                    Core_Entity::factory('Shop_Payment_System',
                    $shop_payment_system_id)
                )
                ->orderParams($_SESSION['hostcmsOrder'])
                ->execute();
            }
            else
            {
                ?><h1>Ошибка! Не указана ни одна платежная система.</h1><?php

            }
            break;
            default:
            ?>
                <form name="order" method="POST" action=".">
                <div id="cart">
                <?php
                    $Shop_Cart_Controller_Show
                    ->couponText(Core_Array::get(Core_Array::get($_SESSION,
                    'hostcmsOrder', array()), 'coupon_text'))
                    ->xsl(
                        Core_Entity::factory('Xsl')->getByName($xslName)
                    )
                    ->show();
                ?>
                </div>
                </form>
            <?php
        }

```

// Блок авторизации пользователя

```

if (Core::moduleIsActive('siteuser') && !SHORTCART_IGNOREUSERS && $step !=
4)
{
    $oSiteuser = Core_Entity::factory('Siteuser')->getCurrent();

    if (is_null($oSiteuser))
    {
        // Авторизация
        $Siteuser_Controller_Show = new Siteuser_Controller_Show(
            Core_Entity::factory('Siteuser')
        );

        $Siteuser_Controller_Show
            ->location(Core::$url['path'])
            ->xsl(
Core_Entity::factory('Xsl')->getByName(Core_Array::get(Core_Page::instance()
->libParams, 'userAuthorizationXsl'))
        )
            ->show();

        // Регистрация
        $Siteuser_Controller_Show = new Siteuser_Controller_Show(
            Core_Entity::factory('Siteuser')
        );

        $Siteuser_Controller_Show->xsl(
Core_Entity::factory('Xsl')->getByName(Core_Array::get(Core_Page::instance()
->libParams, 'userRegistrationXsl'))
        )
            ->location(Core::$url['path'])
            ->fastRegistration(TRUE)
            ->properties(TRUE)
            //->showMaillists(TRUE)
            ->show();
    }
}

```