

# Описание интеграции модуля в страницы сайта

## Форма регистрации пользователя

Варианты интеграции:

- Использовать шаблон компонента из модуля.

Существуют следующие шаблоны с уже интегрированной бонусной системой:

- компонент "bitrix:system.auth.registration", шаблон "flat\_pb"
- компонент "bitrix:system.auth.registration", шаблон ".default\_pb"
- компонент "bitrix:main.register", шаблон ".default\_pb"

При использовании компонента "bitrix:system.auth.registration", необходимо выбрать шаблон скопировать перед использованием, указывая название то, что указано в настройке "Шаблон системных компонентов авторизации (system.auth.\*)" в главном модуле.

- Разместить компонент "efox.premiumbonus:phone.verify.form" (форма подтверждения телефона) в собственный шаблон компонента.
- Интегрировать через JS API в собственный шаблон компонента.

## Форма изменения профиля пользователя

Варианты интеграции:

- Использовать шаблон компонента из модуля. Существует следующий шаблон с уже интегрированной бонусной системой:
  - компонент "bitrix:main.profile", шаблон ".default\_pb"
- Разместить компонент "efox.premiumbonus:phone.verify.form" (форма подтверждения телефона) и "efox.premiumbonus:balance" (информация о бонусном балансе) в собственный шаблон компонента
- Интегрировать через JS API в собственный шаблон компонента.

## Форма оформления заказа

Варианты интеграции:

- Использовать шаблон компонента из модуля. Существует следующий шаблон с уже интегрированной бонусной системой:
  - компонент "bitrix:sale.order.ajax", шаблон "bootstrap\_v4\_pb"
- Разместить компоненты "efox.premiumbonus:phone.verify.form" (форма подтверждения телефона) и "efox.premiumbonus:bonus.writeoff" (форма использования ) в собственный шаблон компонента.
- Интегрировать через JS API в собственный шаблон компонента.

# Описание JS API

## Подключение JS API

```
<?php
    Bitrix\Main\UI\Extension::load(["efox.premiumbonus.api"]);
?>
```

## Форма Подтверждения телефона.

### Инициализация формы подтверждения телефона:

```
<!--Блок с кнопкой “Подтвердить”-->
<div id="efpbVerifyContainer"></div>
<script>
    BX.ready(function () {
        BX.Efox.PremiumBonus.Api.initVerifyForm({
            // Параметр необязательный. Jquery-селектор с input`ом телефона
            phoneSelector: "[name=USER_PHONE_NUMBER]",
            // Параметр необязательный. Для кастомизации вывода результатов
            // расчета используйте JS-события, описанные ниже
            container: "#efpbVerifyContainer",
            // Параметр необязательный. Если true, то значение с input`а
            // phoneSelector будут использованы в форме подтверждения телефона
            updateOnChange: true,
            // Параметр необязательный. Если true, то после подтверждения
            // указанный номер телефона будет вставлен в input телефона, указанный в
            // параметре phoneSelector
            updateOnVerified: true,
        });
    });
</script>
```

### Открыть всплывающую форму:

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.OPEN, [{phone: "71234567890",
disabled: true}]);
```

или

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.OPEN);
```

Параметры события:

- phone - Номер телефона по умолчанию, который будет использован во всплывающей форме;
- disabled - Флаг, может ли пользователь изменить номер во всплывающей форме

## Обработка успешного подтверждения телефона:

```
BX.addCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.VERIFIED, function(params) {  
    console.log("Подтвержден телефон ", params.phone);  
});
```

## Установить номер телефона в форму подтверждения:

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.SET_PHONE, [{phone:  
    "71234567890", disabled: true}]);
```

Параметры события:

- phone - Номер телефона по умолчанию, который будет использован во всплывающей форме;
- disabled - Флаг, может ли пользователь изменить номер во всплывающей форме

## Закрыть всплывающую форму:

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.CLOSE);
```

## Обработка отправки СМС с кодом подтверждения:

```
BX.addCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.SENTED, function(params) {  
    console.log("На номер телефона ", params.phone, " было отправлено СМС с кодом  
    подтверждения");  
});
```

## Обработка события, что телефон уже был подтвержден, что форма подтверждения телефона не будет показана:

```
BX.addCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.VERIFY.ALREADY_VERIFIED, (params) => {  
    console.log("Телефон уже подтвержден" + params.phone);  
});
```

# Форма Использования баллов в оформлении заказа.

## Инициализация формы использования баллов:

```
<!--Блок с результатами расчета по списанию бонусов и кнопкой "Списать баллы"-->
<div id="efpb0rderContainer"></div>
<script>
BX.ready(function () {
    BX.Efox.PremiumBonus.Api.initOrder({
        container: "#efpb0rderContainer", // Параметр необязательный.
    Для кастомизации вывода результатов расчета используйте JS-события,
    описанные ниже
    });
    });
</script>
```

## Открыть форму использования баллов:

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.ORDER.OPEN);
```

Если телефон уже подтвержден, то будет открыта форма использования баллов. Иначе будет открыта форма подтверждения телефона, и уже после нее будет автоматически открыта форма использования баллов.

## Обработка результатов расчета списываемых и начисляемых бонусов:

```
BX.addCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.BONUS_WRITE_OFF.UPDATED,
function(params) {
    console.log(params);
});
```

Параметры события:

- `bonus` - объект с данными по расчету бонусов. Например:

```
{
    error: "Невозможно списать 123 бонусов. Доступно 0", // текст ошибки при попытке списать, например, несуществующие баллы
    data: {
        external_purchase_id: "",
        items: [
            {
                amount: 1399,
                payment: 399,
```

```

        discount_external: 0,
        discount_premiumbonus: 0,
        write_off_bonus: 1000,
        external_item_id: "119"
    }
],
total_write_off_bonus: 1000, // списываемое кол-во бонусов за заказ
total_write_on_bonus: 19.95, // начисляемое кол-во бонусов за заказ
total_card_payment: null,
total_discount_external: 0,
total_discount_premiumbonus: 0,
balance: 19.95, // бонусный баланс после списания баллов
order_code: null
}
}

- total - объект с данными по итоговым суммам (и скидкам) по заказу. Например:
{
DISCOUNT_BONUS_PRICE: 1000, // Скидка бонусами
DISCOUNT_BONUS_PRICE_FORMATED: "1 000 руб.", // Скидка бонусами форматированная
DISCOUNT_PRICE: 1000, // Скидка общая
DISCOUNT_PRICE_FORMATED: "1 000 руб.", // Скидка общая форматированная
ORDER_TOTAL_PRICE: 899, // Итого сумма к оплате
ORDER_TOTAL_PRICE_FORMATED: "899 руб.", // Итого сумма к оплате форматированная
}

```

## Информация о бонусном балансе текущего пользователя

Инициализация блока о бонусном балансе:

```

<!--Блок с информацией о бонусном балансе текущего пользователя-->
<div id="efpbBalanceContainer"></div>
<script>
BX.ready(function () {
    BX.Efox.PremiumBonus.Api.initBalance({
        container: "#efpbBalanceContainer", // Параметр необязательный.
Для кастомизации вывода результатов расчета используйте JS-события,
описанные ниже
    });
});
</script>

```

Открыть блок с информацией о бонусном балансе:

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.BALANCE.OPEN);
```

Если телефон подтвержден, то будет открыт блок с информацией о бонусном балансе

## Обновить информацию о бонусном балансе:

```
BX.onCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.BALANCE.RELOAD);
```

## Обработка ошибки вывода бонусного баланса:

```
BX.addCustomEvent(BX.Efox.PremiumBonus.Api.EVENTS.BALANCE.ERROR, function(errors) {  
    console.error("efox:premiumbonus balance load error: ", errors);  
});
```