Reward Points 1.1.16

Getting Started

Welcome to the **Reward Points Documentation**. Whether you are new or an advanced user, you can find useful information here. First of all we recommend to check the following links:

- How to install extension
- Configuration settings

How to install extension

1. Login to Magento backend.
2. Go to **System > Tools > Compilations**.

If Compiler Status is **Enabled**, disable the compilation by clicking the button **Disable** in the upper right corner.

1. Backup your store database and web directory.
2. Download extension from your Personal Account.
3. Unzip the extension locally.
4. Copy contents of the folder **step1** to the root directory of Magento store.
5. Copy contents of the folder **step2** to the root directory of Magento store.
6. Completely clear the site cache.

Go to **System > Cache Management**. Click button **Flush Magento Cache**. If you want to clear the cache manually, you need to remove a folder /var/cache at the server.

1. Logout from Magento backend and login again.
2. Go to **System > Configuration > Mirasvit Extensions > System**, at form **Installed Extensions** press the button **Run validation tests for all extensions** for check extension installation and configuration.
3. If compilation was enabled before installation, you need to run recompilation process to enable compilation. Go to **System > Tools > Compilation** and click on **Run Compilation Process**.

If you did not disable compilation and store is down after installation, follow the link and make next steps to recover the store.

1. Setup a cron job for Magento if you don't have it yet. See How to Setup Cron for Magento.

Next steps

- Configure extension
- Create Earning Rules
- Create Spending Rules
- Create Notification Rules

How to Setup Cron for Magento

How to upgrade extension

To upgrade extension follow next steps:

1. Login to Magento backend.
2. Go to **System > Tools > Compilations**. If Compiler Status is **Enabled**, disable the compilation by clicking the button **Disable** in the upper right corner.
3. Backup your store database and web directory. If you have customizations, please, create a separate backup of **app/code/local/Mirasvit** before upgrading to the new version.
4. Download new extension package from your Personal Account.
5. Unzip the extension locally.
6. Temporary disable extension. Check Temporary Disabling.
7. Copy contents of the folder **step1** to the root directory of Magento store.
8. Copy contents of the folder **step2** to the root directory of Magento store.
9. Completely clear the site cache. Go to **System > Cache Management**. Click button **Flush Magento Cache**. If you want to clear the cache manually, you need to remove a folder **/var/cache** at the server.
10. Logout from Magento backend and login again.
11. If compilation was enabled before upgrade, you need to run recompilation process to enable compilation. Go to **System > Tools > Compilation** and click on **Run Compilation Process**.

If you did not disable compilation and store is down after upgrade, follow the link and make next steps to recover the store.

How to temporary disable extension

To temporary disable the extension follow the next steps:

1. Login in to Magento backend, go to System > Tools > Compilation. Disable Compilation Mode (if enabled).
2. Rename the following files:
    - app/etc/modules/Mirasvit_Rewards.xml to app/etc/modules/Mirasvit_Rewards.xml.bak
    - app/etc/modules/Mirasvit_RewardsSocial.xml to app/etc/modules/Mirasvit_RewardsSocial.xml.bak
3. If you have only one extension from Mirasvit installed, remove the following file:

    - app/etc/modules/Mirasvit_MstCore.xml to app/etc/modules/Mirasvit_MstCore.xml.bak
4. Go to System > Cache Management. Press the button Flush Magento Cache.

Note: To enable the extension you need to rename the files back and flush the cache.

How to remove the extension

To remove the extension follow the next steps:

1. Login in to Magento backend, go to System > Tools > Compilation. Disable Compilation Mode (if enabled).
2. Remove the following file:
    - app/etc/modules/Mirasvit_Rewards.xml
    - app/etc/modules/Mirasvit_RewardsSocial.xml
3. If you have only one extension from Mirasvit installed, remove the following file:
    - app/etc/modules/Mirasvit_MstCore.xml
4. Go to System > Cache Management. Press the button Flush Magento Cache.
5. Remove the following folders:
    - app/code/local/Mirasvit/Rewards
    - app/code/local/Mirasvit/RewardsSocial
6. Remove templates and layout files of the extension from your theme folder. The list of such files can be collected from files in the extension package.
7. Run following SQL Query (before running, replace [db_table_prefix] to your database table prefix. It can be found in the file /app/etc/local.xml)

SET foreign_key_checks = 0; DROP TABLE [db_table_prefix]m_rewards_balance; DROP TABLE [db_table_prefix]m_rewards_earning_behavior; DROP TABLE [db_table_prefix]m_rewards_earning_rule; DROP TABLE [db_table_prefix]m_rewards_earning_rule_customer_group; DROP TABLE [db_table_prefix]m_rewards_earning_rule_product; DROP TABLE [db_table_prefix]m_rewards_earning_rule_website; DROP TABLE [db_table_prefix]m_rewards_notification_rule; DROP TABLE [db_table_prefix]m_rewards_notification_rule_customer_group; DROP TABLE [db_table_prefix]m_rewards_notification_rule_website; DROP TABLE [db_table_prefix]m_rewards_purchase; DROP TABLE [db_table_prefix]m_rewards_rate; DROP TABLE [db_table_prefix]m_rewards_rate_customer_group; DROP TABLE [db_table_prefix]m_rewards_referral; DROP TABLE [db_table_prefix]m_rewards_spending_rule; DROP TABLE [db_table_prefix]m_rewards_spending_rule_customer_group; DROP TABLE [db_table_prefix]m_rewards_spending_rule_website; DROP TABLE [db_table_prefix]m_rewards_transaction; SET foreign_key_checks = 1; DELETE FROM [db_table_prefix]core_resource WHERE code='mst_rewards_setup';

Configuration Settings

Go to the **Reward Points > Settings**. You'll find the following settings:

General Settings

Option	Description
Point Unit Name	Name of unit for points (e.g. Reward Point(s), Gold Coin(s)).
Reward points expire after, days	Number of days after which points will be expired. If you don't need this feature, leave this field empty.
Approve earned points on invoice	If enabled, earned points will be added to the customer account after creating a new invoice for order.
Approve earned points on shipment	If enabled, earned points will be added to the customer account after creating a new shipment for order.
Cancel earned points after order refunding	If enabled, earned points will be canceled after creating a new credit memo for order.
Restore spent points after order refunding	If enabled, spent points will be restored after creating a new credit memo for order.
Allow to earn points from shipping charges	If enabled, earned points will be added to the customer account from shipping charges.
Allow to spend points for shipping charges	If enabled, customer is able to spend earned points from shipping charges.
Allow to place orders with grand total = 0	If enabled, customer is able to place orders and earn points with grand total = 0.
Use points and coupons at the same time	If enabled, allows to use discount codes and reward points at the same time. To disable this feature, select one of these options: Disabled with warning - shows warning message about restriction of using points and coupons at the same time. Disabled with block hide - if one of the discounts is applied, another discount block hides.

Notification Settings

Option	Description
Email Sender	Email which will be used to send email messages to customers.
Balance update Email	Template for balance update notification.
Points expire Email	Template for points expiration notification.
Send notification before points expiring (in days)	Number of days before which extension will send points expiration notification to customer.

To change email templates go to **System > Transactional Emails** and customise required templates.

Referrals

Option	Description

Enable Referral System	Enables referral system.
Invitation Email Template	Template for invitation of referrals.

Next steps

- Create Earning Rules
- Create Spending Rules
- Create Notification Rules

Social Networks Settings

Extension allows customers to earn Points by promoting store products and categories via social networks. Go to: **System** > **Confguration** > **Social Networks Settings**. Set up the following settings:

Facebook Settings

- **Show Facebook Like button** - If option enabled, Facebook Like button will be shown on the product and category pages.
- **Facebook App ID** - To receive App ID, please, create an application here https://developers.facebook.com/apps.
- **Facebook API Version** - You can find API Version in settings of your application, for more details please read https://developers.facebook.com/docs/apps/versions.

Twitter Settings

- **Show Twitter Follow button** - If option enabled, Twitter Follow button will be shown on the product and category pages. This option also requires Twitter Application Setup.

Google Plus Settings

- **Show Google Plus +1 button** - If option enabled, Google Plus +1 button will be shown on the product and category pages.

Pinterest Settings

- **Show Pinterest pin button** - If option enabled, Pinterest pin button will be shown on the product page.

When appropriate buttons are shown on product pages, it's' need to create corresponding rules for each social button. To make such rule, go to the **Reward Points -> Earning Rules** and create a rule of **Behaviour** type, with one of these Events:

- **Facebook Like**

- **Twitter Tweet**
- **Google+ Like**
- **Pinterest Pin**

More information on creating an earning rule, can be found at Create Earning Rules section. **Next steps**

- Create Earning Rules
- Create Spending Rules
- Create Notification Rules

Twitter Application Setup

New Twitter API does not allow to instantly check, whether tweet was actually posted by customer. Instead of that, it allows just to register his intention to make a tweet. So, to ensure, that Twitter points was awarded for actual sharing, we need to check existence of actual Twitter post. This is when special Twitter App come, which should be registered at Twitter account of given store. To create one, go to the Twitter Developer Site and log in using your Twitter Account credentials. There, in footer menu proceed to **Tools -> Manage Your Apps**. Click on **Create New App** button and fill the following fields of creation form:

- **Name** - unique name of your application.
- **Description** - short description.
- **Website** - your store's root URL.

Mirasvit Demo Application Setup

Settings, therefore, should look like this example: Image not found https://mirasvit.com/doc/extension_rewards/current/index.php/images/twitter_application_settings.p . After creating of application, open its properties, and jump to the **Keys and Access Tokens** tab. There you will see two parameters, needed to bind our extension with your Twitter App:

- **Consumer Key (API Key)**
- **Consumer Secret (API Secret)**

Copy these keys to the fields of the same name, found at **Reward Points -> Settings -> Social Network Settings** under **Twitter Settings**. **Next steps**

- Create Earning Rules
- Create Spending Rules
- Create Notification Rules

Earning Rules

Earning Rules define ways for customers to earn points. To create a new Earning Rule, go to the **Reward Points > Earning Rules** and press the button **Add New**. Fill in the following fields:

General Information (common for all types of rules)

- **Rule Name** - name of the rule. Used only for internal purposes.
- **Type** - type of the rule. There are following types of rules:
    - **Behaviour Rule** - points can be earned for customer's actions.
    - **Product Rule** - points can be earned for buying specific products.
    - **Cart Rule** - points can be earned for cart amount or contents.
- **Is Active** - rule status.
- **Active From**, **Active To** - period of rule activity.
- **Websites** - websites where rule will be active.
- **Customer Groups** - customer groups for which rule can be applied.
- **Stop further rules processing** - if enabled, there are no more rules will be applied after this rule.
- **Priority** - order which will be used for rules sorting before applying.

Each type of rule have it's own purpose and use scope.

- **Behaviour rule**. It triggers on certain events, that resemble customer's actions - signing in, or writing a review for example.
- **Product rule** is the most simple of them. It features a hint on product pages (and on a categories - it, however, depends from theme), which shows, how much customer will receive points for buying that product. At the cart and checkout stage this rule calculates points amount only from product prices, and includes only tax, included to that price. Shipping charge and separate taxes are skipped.
- **Cart rule**. It does not feature hint on product page, because is applied only to cart in a whole. Therefore, points amount in this rule type is calculated from subtotal - e. q. actually paid money amount, which can include shipping fees and taxes.

Behaviour Rules

Award points to customer for their actions.

Conditions

- **Event** - customer's action which can trigger the rule. Available actions are:
    - **Rates a product**
    - **Signs up**
    - **Logs in**
    - **Votes in poll**
    - **Sends product to friend**
    - **Signs up for a newsletter**
    - **Tags a product**
    - **Writes a review**
    - **Places an order**
    - **Has a birthday**
    - **Is not active for long time**
    - **Referral signs up to website**

    - **Referral makes first order**
    - **Referral makes any order**
    - **Likes a page with Facebook**
    - **Tweets a page with Twitter**

Note: Events "has a birthday" and "is not active for long time" are executed using crontask, once a day.

- **Conditions** - customer's properties, that can be used to make rule application more precise. Available properties are:
    - **Group**
    - **Lifetime sales**
    - **Number of Orders**
    - **Is subscriber to newsletter**
    - **Is referral** (e. q. whether he is invited to the store by other customer)
    - **Is referee** (e. q. whether he invited to the store one or more other customers)
    - **Number of Reviews**
    - **Balance points amount**
    - **Number of referred friends signups**
    - **Number of referred friends orders** - number of orders, placed by customers
    - **Sum of referred friends signups** - total money amount of orders, placed by customers
    - **Number of referred friends ordered at least once**

Actions

- **Number of points to give** - number of points that customer receive if rule is applied.
- **Maximum number of earned points for one customer per day** - limit the number of points gained by customer per day.
  If the limit is exceeded, points will not be added.
- **Assign customer to Group** - allows to automatically transfer customer to desired Group on rule triggering. It is extremely used for planning Threshold Rules (e. q. when customer reaches a certain amount sales, for example) and Referral Policy.

For example, you can create the following behaviour rules in your store:
- Write a review and receive 10 points
- Refer a friend and receive 10 points when your friend makes first order

Product Rules

Award points for buying particular product. This type of rule is recommended, when products are presumably simple and have fixed prices.

Conditions

Can contain either single condition, or their combination. Rule can be applied, only if all of them are met.
- **Conditions** - product's properties, that can be used to make rule application more precise. Available properties are:
    - **Category**

  - **Product Type** (e. q. Simple, Virtual, Bundle, etc.)
  - **SKU** - can contain one or more SKU's. When use one SKU condition should use "is" and "is not" operators, when more than one - "is one of" or "is not one of".
  - **Attribute Set** - product attribute setup, created at **Catalog -> Attributes -> Attribute Sets**
  - **Quantity**
  - **Base Price** - base price of the product, set at the General tab in product's properties
  - **Special Price** - special price, used for actions and sales
  - **Final Price** - actual price, calculated when Tier of Custom prices are used
  - **Base Image**
  - **Base Image Size**
  - **Small Image**
  - **Small Image Size**
  - **Thumbnail**
  - **Thumbnail Size**
  - **PHP Condition**

"PHP Condition" used, when need to filter products on a specific property - custom attribute, for example. Having attribute code, it is possible to get it by putting direct PHP expression in this condition's text field. Here is the universal expression for any product attribute: $object->getAttributeText([ATTRIBUTE_CODE]) == [ATTRIBUTE_VALUE] Attribute codes can be seen at **Catalog -> Attributes -> Manage Attributes**. For example, to get attribute with code "manufacturer" and check, if it is equal "Adidas", we can write down: $object->getAttributeText('manufacturer') == 'Adidas' Also, other product's attributes can be retrieved using this special condition, if the corresponding method exists in Magento API. Quick reference can be found there: Mage_Catalog_Model_Product

Actions

- **Customer Earning Style** - algorithm to calculate the number of earned points.
  - Give X points to customer
  - Give points amount, equal to X percent of price
  - For every Y, give X points
- **Number of Points (X)**
- **Step (Y)**
- **Maximum Distributed Points** - maximum number of possible earned points for current rule.

For example, you can create the following product rules in your store:

- Buy product A and receive 100 points.
- Buy any product with price above $100 and receive 100 points.
- Buy any product in the category Electronics and receive 1 Reward Point for each spent $1.

Cart Rules

Award points for the total amount of actually spend money at the cart. This type of rule is recommended, when using tier or special prices (with taxes, dependent from shipping country, for example), which can dynamically alter the actual product sell price.

Conditions

Can contain either single condition, or their combination. Rule can be applied, only if all of them are met.

- **Conditions** - cart's properties, that can be used to make rule application more precise. Available properties are:
    - **Subtotal**
    - **Total Items Quantity**
    - **Total Weight**
    - **Payment Method**
    - **Shipping Method**
    - **Shipping Postcode**
    - **Shipping Region**
    - **Shipping State/Province**
    - **Shipping Country**
    - **Coupon Used** - whether any Coupon Code was used by customer on this cart
    - **Coupon Code** - whether particular Coupon Code was used by customer on this cart
    - **Discount Amount** - whether some other discount was applied by customer on this cart
    - **Current Store** - whether cart was created on particular storeview
    - **Current Currency** - whether cart was created using particular Currency

Cart Rule also allows using current customer's properties as Conditions. They are identical to that described above in **Behaviour Rules** section.

Actions

- **Customer Earning Style** - algorithm to calculate the number of earned points.
    - Give X points to customer
    - Give X points for every spent Y
    - Give X points for every Z quantity
- **Number of Points (X)**
- **Step (Y)**
- **Quantity Step (Z)**
- **Maximum Distributed Points** - maximum number of possible earned points for current rule.

For example, you can create the following cart rules in your store:

- Buy 3 products within one order and receive 100 points.
- For every spent $1, receive 1 point.
- For every ordered product, receive 100 points.

Notifications

Each earning rule has additional **Notifications** tab, where a history message can be specified, and a special email notification message, that will be sent to customer, if this particular rule applies. **Message in the rewards history** field should contain a message, that will accompany corresponding **Transaction** , which is created on rule application, and can be seen in field **Comment**. This message can contain special variables, to make this message more personal:

- {{customer_name}} - translates to a current customer's name
- {{referred_customer_name}} - translates a name of a referral, if rule of referral used

These variables works **only** in History Messages. **Message for customer notification email** field can contain an email template, which will be used for email notification about points awards. It also can contain variables, but the same, that used in regular transactional emails.

Spending Rules

Spending Rules define ways for customers to spent points. To create a new Spending Rule, go to the **Reward Points > Spending Rules** and press the button **Add New**. Fill in the following fields:

General Information (common for all types of rules)

- **Rule Name** - name of the rule. Used only for internal purposes.
- **Is Active** - rule status.
- **Active From**, **Active To** - period of rule activity.
- **Websites** - websites where rule will be active.
- **Customer Groups** - customer groups for which rule can be applied.
- **Stop further rules processing** - if enabled, there are no more rules will be applied after this rule.
- **Priority** - order which will be used for rules sorting before applying.

Conditions

Can contain any cart conditions. Rule can be applied, only if conditions are valid. They are set in **Actions** tab, under **Apply the rule only to cart items matching the following conditions** section. Additionally, Spending Rules can utilize as Conditions current customer's properties and some additional attributes of current Cart:

- **Group**
- **Lifetime sales**
- **Number of Orders**
- **Is subscriber to newsletter**
- **Is referral** (e. q. whether he is invited to the store by other customer)
- **Is referee** (e. q. whether he invited to the store one or more other customers)
- **Number of Reviews**
- **Balance points amount**
- **Number of referred friends signups**
- **Number of referred friends orders** - number of orders, placed by customers
- **Sum of referred friends signups** - total money amount of orders, placed by customers
- **Number of referred friends ordered at least once**
- **Coupon Used** - whether any Coupon Code was used by customer on this cart
- **Coupon Code** - whether particular Coupon Code was used by customer on this cart
- **Discount Amount** - whether some other discount was applied by customer on this cart

- **Current Store** - whether cart was created on particular storeview
- **Current Currency** - whether cart was created using particular Currency

These additional conditions, listed above, can be set on separate **Conditions** tab.

Actions

- **For each spent X points**
- **Customer receive Y discount**
- **Spend minimum**
- **Spend maximum**

NOTE: To allow customer spend his points, you must create at least one spending rule.

Notification Rules

Notification Rules allow to setup different messages for customers, which notify customers about possible earning/spending points. To create a new Notification Rule, go to the **Reward Points > Notification Rules** and press the button **Add New**. Fill in the following fields:

General Information (common for all types of rules)

- **Rule Name** - name of the rule. Used only for internal purposes.
- **Is Active** - rule status.
- **Active From**, **Active To** - period of rule activity.
- **Websites** - websites where rule will be active.
- **Customer Groups** - customer groups for which rule can be applied.
- **Stop further rules processing** - if enabled, there are no more rules will be applied after this rule.
- **Priority** - order which will be used for rules sorting before applying.

Conditions

Can contain any cart conditions. Rule can be applied, only if conditions are valid.

Actions

- **Show message on** - list of pages, where we would like to show this notification message.
- **Message** - text of message to show.

In messages you can use two special variables - [earn_points] and [spend_points], which display currently earned and spent points.

Known Issues

I had set up Earning rule, but points calculations does not

include tax

This issue can appear due either to wrong rule setup, or Magento's tax calculation details. **Solution:**

- Make sure, that you're using **Cart** type for the earning rule. Only this type can calculate points from actually paid amount;
- Make sure, that tax is applied before discount. This option is set at **System -> Configuration -> Sales -> Tax -> Calculation Settings -> Apply Customer Tax**. If there's **After discount** value, then our extension will not include tax to points calculation.

On adding products to the Cart or/and proceeding to Cart and Checkout grand total turns to zero

This issue can appear, when you use third-party extension, which alters total recalculation or quote saving event and thus conflicting with our extension. **Solution:** Go to **Reward Points -> Settings -> Advanced**. This section contains options, which allow to disable points recalculation on Magento events, and enable it through observer functions. In most cases you need to set these options:

- **Refresh points on cart and checkout** to **Yes**.
- **Disable Points Refreshing at Stages** to **Frontend quote save** or/and **Order Completion**.

This will disable points refreshing on Magento's events, and make points refresh only on Cart and Checkout stage. If this will not help, try to experiment with disabling refreshing for different stages. One of these combinations will resolve the conflict. See also next issue, that also can apply.

I had installed POS (Point-of-Sales) extension, and orders from POS turned to zero

This issue can appear due to custom shopping cart or custom quote, that POS uses for their order creation. Our extension uses standard ones, therefore, also can conflict. **Solution:** Typically Points-of-Sales extensions features their own URI space with easily-recognized pattern. So in most cases it is enough to exclude these using **Reward Points -> Settings -> Advanced -> Exclude points refresh, when URI request has pattern** option. If Point of Sales extension also modifies cart events, you might also need to disable points refresh at **Backend quote save** at **Reward Points -> Settings -> Advanced -> Disable Points Refreshing at Stages**. **Note:** Although this option shall contain a regular expression, it also can contain a regular string. If POS URI space has very simple pattern - for example, /pos/ or PointOfSales_PointOfSales - just enter it to that field, it will also properly filter POS pages and requests.

IDEV Onepage Checkout (version <= 4.x) does not show correctly the points value

For example, it should show '100 Reward points', but it shows '$100' on the checkout page. **Solution**:

1. Open a file /app/design/frontend/[your theme]/template/onestepcheckout/summary.phtml.

2. Find the line with <?php echo $this->helper('checkout')->formatPrice($value); ?>.
3. Replace that line by the following code: <?php if ($code == 'rewards_earn' || $code == 'rewards_spend'): ?> <?php echo Mage::helper('rewards')->formatPoints($value) ?> <?php else: ?> <?php echo $this->helper('checkout')->formatPrice($value); ?> <?php endif;?>

New Order notification email, when contains Reward Points discount, has untranslated keywords

For example, email subtotal section contains row "Discount (100 Mukafat Xallari)", while should be "Endirim (100 Mukafat Xallari)". **Solution**:

1. Open a file app/locale/[your locale]/Mage_Sales.csv.
2. Find appropriate keyword and give it proper translation.

When I try to filter grid at your extension (Reward Points -> Referrals for example), it crushes with SQL dump.

Check your SQL dump, it shall resemble the following: a:5:{i:0;s:712:"SQLSTATE[42S22]: Column not found: 1054 Unknown column 'CONCAT(`ce1`.`value`, ' ',`ce2`.`value`)' in 'where clause', query was: SELECT COUNT(*) FROM `m_rewards_referral` AS `main_table` LEFT JOIN `customer_entity_varchar` AS `ce1` ON ce1.entity_id=main_table.customer_id LEFT JOIN `customer_entity_varchar` AS `ce2` ON ce2.entity_id=main_table.customer_id LEFT JOIN `customer_entity_varchar` AS `ce3` ON ce3.entity_id=main_table.new_customer_id AND ce3.attribute_id=5 LEFT JOIN `customer_entity_varchar` AS `ce4` ON ce4.entity_id=main_table.new_customer_id AND ce4.attribute_id=7 WHERE (ce1.attribute_id=5) AND (ce2.attribute_id=7) AND (`CONCAT(``ce1```.```value``, ' ',``ce2```.```value``)` LIKE '%arnold%')";i:1;s:6043:"#0 /home/ipet/public_html/includes/src/Varien_Db_Statement_Pdo_Mysql.php(110): Zend_Db_Statement_Pdo->_execute(Array) This issue appears, when used third-party extension **BL Custom Grid** for Grid customization. **Solution:** Unfortunately, this conflict can not be resolved from our side. BL Custom Grid operates on top of our extension, and therefore, overrides our classes. Moreover, this extension is old and abandoned, so the best solution in this case would be either disabling BL Custom Grid, or customizing its code to fit Mirasvit extension.

How To

- **How can I make customer earn points from buying specific products**
- **How can I prevent rules from multiply?**
- **How can I set up trading standard Magento coupons for additional points?**
- **How can I set up rule to award both referral customer, and it's referee?**
- **How can I create so-called 'threshold' and 'milestone rule'?**
- **How can I award customer with 'gift' points, that he can use on his first order?**

- **How can I disable simultaneous Magento's coupon using and spending/earning Reward Points?**
- **How can I disable frontend notification about behavioural points awards?**
- **How can I disable Reward Points spending block at Cart or Checkout?**
- **How can I enable Rich Pins for my store?**
- **How do I disable certain social buttons?**
- **How do I customize social buttons or/and add custom button for unsupported social network?**
- **How can I make social button apply only once per store as a whole?**
- **How can I hide social buttons on Category pages, and display only on Product Pages?**
- **How can I move buttons on Products page?**

How can I make customer earn points from buying specific products

It can be done in two ways - using Product or Cart rules. If you use **Product** rule:

- Jump to **Conditions & Actions** tab, and add one (or several) conditions as follows:
    - If you need to restrict earning to some product Categories, use **Category** condition from **Product Attributes** subsection in drop-down list;
    - If you need to restrict earning to some particular product, use **SKU** condition from the same subsection;
- If you need to filter more than one product or category, then use one of following approaches:
    - Change **ALL** to **ANY** at master condition **If ALL of these conditions are TRUE** (this will make rule trigger, when at least one condition applies), and create a number of conditions for each category or product SKU;
    - Place to the condition comma-separated list of values, and select operator **is one of** instead of **is**.

If you use **Cart** rule:

- If you need to control not only products, but also their quantity:
    - Jump to **Conditions** tab and add a new condition. In drop-down list select **Product subselection**. It will produce nested conditions block with a master condition, which filters quantity of products selected;
    - In this nested block set up quantity, that should trigger the rule. If you need to award customer for buying 3 items of some product - then nested block master condition should contain **If total quantity equals or greater than 3 for a subselection of items in cart matching ALL of these conditions**;
    - Add to this nested block Category, SKU, or other attributes of products, that should trigger the rule - like described for Product rule;
- If you need just to control product presence in the Cart:
    - Jump to **Actions** tab, and create a new condition in **Apply the rule only to cart items matching the following conditions** section;
    - Add appropriate product attributes, that should trigger the rule - as described above.

Note: If you use Cart rule, do not use simultaneously both of described approaches - they will conflict. Create instead two Earning rules - you may also see next case to properly trigger them;

How can I prevent rules from multiply? For example: if there's two rules for products, and customer added to cart both - but only one rule should trigger.

- Jump to **General Information** tab and assign to these rules priorities according to product's promotional significance. Say, you have products A and B, and if they both in cart - reward for B should be applied. Rule for product A thus should have priority 20, and for A - 10;
- Set option **Stop further rules processing** to **Yes**. This will prevent rules with same or lower priority from triggering, even if their conditions apply.

How can I set up trading standard Magento coupons for additional points

- Create an Earning Rule at **Reward Points -> Earning rules**, of **Cart** type;
- Jump to the **Conditions** tab, and make new condition. In drop-down list, in section **Additional Cart Attributes** find **Coupon Code** condition and make it equal to coupon code, that should be traded for points;
- Proceed to **Actions** tab, set **Customer Earning Style** to **Give X points to customer**, and in field **Number of Points (X)** set appropriate quantity reward points, that should be awarded on coupon application;
- Activate the rule (and set activity period, if need).

Note: if you have coupons generated by some algorithm, **Coupon Code** condition also supports different operations - such as substring detection (contains operator) and check for one of coupon array (is one of operator), so it can be flexibly fitted to your coupon policy.

How can I set up rule to award both referral customer, and it's referee (e. q. customer, who invited him).

It can not be done via one earning rule, you will need two of them - one for Referral, and the second for Referee. Rule for Referral can be created this way:

- Create a new **Earning Rule** of any type at **Reward Points -> Spending Rule**;
- Jump to the **Conditions** (or **Conditions and Actions**) tab, and add new condition. In drop-down list, in **Customer** section select **Is Referral** condition and set it to **Yes**;
- Add other conditions - for example, if you need to set awards for certain order number, refer to milestone rule case, below in this section;
- Jump to **Actions** section, set points to earn, save and activate rule.

Rule for Referee can be created only using **Behaviour** type of Earning Rule:

- Create a new **Earning Rule** of **Behaviour** type at **Reward Points -> Spending Rule**, and jump to the **Conditions and Actions** tab;
- If you wish to award customer, when referral registers, pick event **Referred customer signs up in store**;
- If you wish to award customer, when referral places an order, pick event **Order from referred customer**;
- If you also need to award referee for a certain number of order, placed by referral - or make other limitations regarding referral - use conditions from **Referral Customer** subsection. For example, if you need to award points only for first referral's order, you need **Referral: Number of Orders** condition;
- Save and activate rule.

How can I create so-called 'threshold' and 'milestone rule', which will award customer with additional points on reaching sales amount or certain orders number

To award customers with additional points depending of their achievements, you need to use either **Cart** , or **Behaviour** rule. Both of them support conditions, that allows checking of customer's properties. Simply add them to the **Conditions** tab to make rule threshold or milestone. Here is the list of conditions, that can be used for the most cases:

- Award only with first order. Use condition - **Number of Orders** - and set it to **0**. It does not applies only to **Behaviour** rules, which use **Customer places an order** event - in this case you need to set **1** (because this event triggers after order is placed);
- Award customer for certain X-th order (say, 10-th). Use the same **Number of Orders** condition, but it shall be equal to X - 1 (for **Customer places an order** shall be equal to X);
- Award customer for reaching certain X points on his balance. Use two conditions **Balance points amount** - for lower limit, and upper limit (if applies).

- Award customer for spending X money in store. Use two conditions **Lifetime Sales** - for lower limit, and upper limit (if applies).

Not only customer overall achievements can be checked, but particular purchase as well. Use for that additional conditions, located under **Cart Attributes**, for rules of **Cart** type.

How can I award customer with 'gift' points, that he can use on his first order.

Customer can receive points immediately after registration using only **Behaviour** rule. Here is how it can be done:

- Create a **Behaviour** rule at **Reward Points -> Earning points** and on **Conditions & Actions** pick up event **Customer signs up in store**;
- Set **Customer Earning Style** to **Give X points to customer**, and set points amount to receive;
- If you wish to suppress frontend notification for this rule, add **Signing up** event at **Reward Points -> Settings -> Display -> Disable Behaviour Notifications**;
- Add other conditions, if you need to - for example, a check for Referral (see case above);
- Save and activate rule.

This rule will make customer receive points on registration stage, and they can be already spent on first order.

How can I disable simultaneous Magento's coupon using and spending/earning Reward Points

This can be done in two ways - via setting and via rules. First way, totally forbids simultaneous using coupons and Reward Points:

- Go to **Reward Points -> Settings -> General Settings -> Use points and coupons at the same time**;
- If you wish to display warning, select **Disabled with warning**;
- If you wish to hide Reward Points spending block, select **Disabled with block hide**.

Second way, allows to flexibly vary policy depending on coupon type and discounting policy:

- Create a Spending Rule at **Reward Points -> Spending Rules**;
- Jump to the **Conditions** tab, and add one of following condition:
    - **Coupon Used**, with **No** value. Forbids spending, when any coupon is used and hides spending block;
    - **Coupon Code**, with appropriate operator, and coupon code. Allows to forbid spending only when certain code (or their group) is applied.
    - **Discount Amount**. Forbids spending, when certain amount of discount is applied prior to spending (including custom and by third-party extensions). Zero value allows to forbid spending, when any discount is applied, some numeric value establishes a threshold, above (or under) which Reward Points spending is forbidden.
- Jump to **Actions** and set spending rate, as in regular spending rule.

Earning can be forbidden similarly - but only by using rule and conditions, described above.

How can I disable frontend notification about behavioural points awards - for writing a review for example.

All behavioural frontend notifications are governed by **Reward Points -> Settings -> Display -> Disable Behaviour Notifications**. Select there option **Writing a review** (along with other frontend notifications, that you wish to hide) and save configuration.

How can I disable Reward Points spending block at Cart or Checkout

Reward Points spending block can be hidden by setting to **No** options from **Reward Points -> Settings -> Display** section:

- for cart: **Display Reward Points block on cart page**;
- for checkout: **Display Reward Points block on checkout page**

How can I enable Rich Pins for my store

Our extension supports Rich Pins by default. The only thing you need to do - set up special additional meta-tags on your product page, which our extension can use. Here is an official guideline to prepare and enable Rich Pins for your store: Enable Rich Pins for your store

How do I disable certain social buttons

Each social button has its own section at **System -> Configuration -> Mirasvit Extensions -> Social Network Settings**. For example, if you need to hide Facebook button, open **Facebook settings** subsection, and set option **Show Facebook Like button** to **No**. This will hide that button both from product and category pages. The same applies to other social buttons.

How do I customize social buttons or/and add custom button for unsupported social network.

Each social button in fact is a separate template, which is bringed together dynamically. So, to customize social buttons, you need to edit these templates:

- **Facebook**: /[magento_root]/app/design/frontend/base/default/template/mst_rewardssocial/buttons/facebook/like.phtml
- **Twitter**: /[magento_root]/app/design/frontend/base/default/template/mst_rewardssocial/buttons/twitter/tweet.phtml
- **Google+**:

/[magento_root]/app/design/frontend/base/default/template/mst_rewardssocial/buttons/googleplus/one.phtml
- **Pinterest**: /[magento_root]/app/design/frontend/base/default/template/mst_rewardssocial/buttons/facebook/like.phtml
- **Refer friends**: /[magento_root]/app/design/frontend/base/default/template/mst_rewardssocial/buttons/pinterest/pin.phtml

If you need to add additional social button, then you need to alter this template: /app/design/frontend/base/default/template/mst_rewardssocial/buttons.phtml **Important note:** Adding new social button, described here, is merely cosmetic. Points **won't** be awarded for custom buttons, because it will require code modification as well. Contact Mirasvit Team or its partners, if you wish not only to have additional button, but include it to the rewards policy as well.

How can I make Facebook like button (or other social button) apply only once per store as a whole (or for certain page)

Pick up a template for a social button (listed in case above), and add desired URL to these locations:
- **Facebook**: set URL as value to href attribute of fb:like tag;
- **Twitter**: set URL to $currentUrl variable (requires PHP code);
- **Google+**: set URL in RewardsGoogleplusOne javascript constructor (requires PHP code)
- **Pinterest**: replace $this->getEncodedPageUrl() variable with desired URL.

How can I hide social buttons on Category pages, and display only on Product Pages?

For that you will need to edit our social buttons layout. By default it is located here: /[magento_root]/app/design/frontend/base/default/layout/mst_rewardssocial.xml Remove from layout block, marked catalog_category_view, and purge ALL cache to make changes apply. If you use custom layout, then path would be: /[magento_root]/app/design/frontend/[YOUR_PACKAGE]/[YOUR_THEME]/layout/mst_rewardssocial.xml Find values for placeholders in path above in these settings:
- YOUR_PACKAGE - **System -> Configuration -> Design -> Package -> Current Package Name**
- YOUR_THEME - **System -> Configuration -> Design -> Themes -> Layout**. If there is empty value, than theme's name is default.

How can I move buttons on Products page.

Social buttons can be moved in two ways, which are dependant from page layout, that you use: **Method 1: move by layout**:

1. Open our social buttons layout, which is located here by default: /[magento_root]/app/design/frontend/base/default/layout/mst_rewardssocial.xml

2. Find block, marked catalog_product_view, and in child element <reference name="content"> replace name attribute to layout block, **under** which buttons should appear. Typical locations are:
    - **product.info.additional** - displays buttons after additional info block of product
    - **product.info.media** - displays buttons under image of product
    - **product.info.addtocart** - displays buttons under **Add To Cart** button

   Valid block names can be seen in layout file below (by default) in section catalog_product_view:
    - /[magento_root]/app/design/frontend/base/default/layout/catalog.xml

   Or, if you use custom theme or custom layout:
    - /[magento_root]/app/design/frontend/[YOUR_PACKAGE]/[YOUR_THEME]/layout/catalog.xml
    - /[magento_root]/app/design/frontend/[YOUR_PACKAGE]/[YOUR_THEME]/layout/page.xml

   Refer to previous case, how to know your package and theme names.
3. In some cases, when Product page has only content block, or has complex layout, which requires additional buttons adjusting, proceed to next method:

Method 2: move by CSS:

1. Open our CSS stylesheet, located by default here: /[magento_root]/skin/frontend/base/default/css/mirasvit_rewards.css
2. Find block, marked .rewardssocial-buttons, and put there additional styles, that precisely reposition our buttons to the proper place.
3. Purge ALL cache to make changes apply.

In some cases mobile theme requires different position of buttons, while regular theme should remain the same. In this case you need to reposition buttons using styles by adding media section to our stylesheet. E. q. you need to add code like this, where max-width parameter defines maximum resolution, which triggers mobile layout: @media only screen and (max-width: 640px) { .rewardssocial-buttons { top: 100px; left: 10px; } } Change Log

Change Log

1.1.32

(2018-09-12) Fixed

- Rule's condition "Category" does not work for categories with root not equal to 1
- Issue "Points return only for the first credit memo of the order"
- Points for first order earns twice
- Compatibility with Amasty_Scheckout 3.1.3
- Compatibility with Simplecheckout
- Fixed Notice: Undefined offset: 1 in Mirasvit/Rewards/controllers/Adminhtml/Rewards/Earning/RuleController.php
- Option "Use points and coupons at the same time" does not work for Amasty Checkout
- Wrong points calculations on product page when "Display Product Prices In Catalog" set to "Including Tax" and "Apply Discount On Prices" set to "Excluding Tax"
- Points does not recalculate after applying coupon
- Custom order numbers have wrong url on transaction page
- Signup notification shows on account page

- Compatibility with Smartwave Ajaxcart (after signup cart total is 0 for the first added product)
- "Spend maximum" does not work correct in Amasty Scheckout v3.2.6

1.1.31

(2018-09-12) Fixed
- Issue when other extensions override twitter's window.onload function
- Unlike not working for urls with question mark
- Compatibility with Amasty_Scheckout 3.1.3
- Compatibility with Idev_OneStepCheckout ajaxed shipping and payment methods
- Spending Rule max points calculations
- Earned points do not cancel on order cancelation

1.1.30

(2018-05-30) Fixed
- Syntax error, unexpected '['

1.1.29

(2018-05-22) Fixed
- Compatibility with Amasty_Scheckout - 2.9.10
- Compatibility with Magestore_Onestepcheckout
- Compatibility with Lotusbreath_Onestepcheckout - 4.2.0
- Behavior rule conditions allows to select product conditions
- Column 'increment_id' in where clause is ambiguous
- Product's attributes for product rules
- Number of reviews include only approved reviews

1.1.28

(2018-04-06) Fixed
- Rewards calculation for shipping
- Issue when notification displays twice on login page
- Compatibility with lotusbreath_onestepcheckout-4.2.0
- Issue when tax applied twice to Earning points

1.1.27

(2018-02-22) Fixed
- Points calculation for shipping
- Date saving in non USA locale

1.1.26

(2018-02-14) Fixed
- Points are not calculating for bundle products

- Awarding not canceled when store uses SID in the url
- After deleting product from cart points are not updating correctly for cart rule
- Error "Invalid App ID: The provided app ID does not look like a valid app ID." when Facebook application become invalid
- Issue when discount is not updating after currency switching

1.1.25

(2017-12-26) Improvements
- Referral Invitation form on product page

Fixed
- Added settings to configure displaying of Social Buttons block

1.1.24

(2017-12-21) Fixed
- Compatibility with IWD Onepagecheckout

1.1.23

(2017-12-20) Fixed
- Compatibility for Notification Rules with IWD Onepagecheckout

1.1.22

(2017-12-13) Improvements
- Styles for TM_FireCheckout

1.1.21

(2017-12-11) Fixed
- Compatibility with php 5.3

1.1.20

(2017-12-07) Fixed
- Compatibility with FireCheckout
- Compatibility with MageWorx InstantCart

1.1.19

(2017-12-05) Features
- Added variables for Notification Rules
- Added Portuguese (Brazil) locale

Improvements
- Compatibility with Amasty_Shiprestriction

Fixed

- Compatibility with Aheadworks onestep checkout
- Multiple spending rules limit calculation
- Validator

1.1.18

(2017-09-26) Features
- Added possibility to automatically assign a Group to customer on Behaviour Rule triggering

Fixed
- Some minor backward compatibility issues

1.1.17

(2017-09-19) Features
- Added earning points from Facebook Share

Fixed
- Discount amount calculation for shipping
- Incorrect notification display depending on customer group
- Incorrect comment when more than one backend locale used
- Discount Amount percent value calculation

1.1.16

(2017-06-02) Features
- Allowed conflict resolution using Advanced section in Settings (#108)
- Allowed display/hide behaviour frontend notifications (#108)
- Improved documentation, added How-to and Known Issues sections (#109)

1.1.15

(2017-04-21) Features
- Allowed points spending for backend orders (#101)
- Added options to display/hide rewards block on checkout and cart pages (#99)
- Added condition to check for email confirmation in (#96)

Fixed
- Fixed negative MIN and MAX for rules (#98)
- Fixed delete purchases on quote delete

1.1.14

(2017-01-05) Features
- Added variables to transaction comments (#92)
- Made order links in transaction comments clickable (#88)

- Enabled points hint display for grouped and bundled products (#87)
- Added Email condition to the both earning and spending rules (#83)
- Added promotional messages for behaviour rules Sign Up and Write Review (#75)

Fixed
- Restored and improved Magestore Onestepcheckout support (#90)
- Fixed multiple category equal check in earning rules (#85)
- Fixed standard RWD checkout support (#81)
- Fixed bug with pagination in customer's transactions grid (#78)
- Fixed Refer Friends share link generation for uncategorized products (#77)
- Fixed incorrect check for Special Price in Product Rule (#73)
- Removed obsolete RWP-109 bug check, that makes validator return false fail (#71)
- Fixed tax amount missing in earning rules (#62)
- Fixed disabling coupons and rewards in Firecheckout (#59)

1.1.13

(2016-10-11) Features
- Allowed points calculation as percent from product price (#56)
- Added Store and Currency conditions to Earning and Spending rules (#54)
- Added Coupon Code condition to the Earning Rule (#52)

Fixed
- Improved documentation regarding rules conditions (#57)

1.1.12

(2016-10-11) Fixed
- Fix issue with incorrect points calculations for grouped products when Product Rule is used (#50)
- Fixed issue with awarding points below minimal spending limit (#50)
- Fixed issue with "Is Referral" condition, which does not work with event "Customer signs up" (#50)
- Made "Is Referral" and "Is Referee" conditions Yes/No dropboxes (#50)
- Fixed store crush issue when Rewards Social Buttons block is called directly (#46)
- Made refer links redirect to SEO-friendly URLs (#44)
- Fixed incorrect calculation of minimal points for spending (#42)
- Fixed Ultimo theme compatibility (#39)
- Fixed firecheckout styles (#37)

Features
- Added support of Aitoc Onestepcheckout (#49)
- Added to Spending Rules filter by customer attributes (#50)

1.1.10

(2016-08-11) Fixed

- Issue when points are not earned for pins (#30)

1.1.9

(2016-08-05) Fixed
- Issue when spending rules discount used only base currency (#22)
- Sign In points awarding when save process is altered by checkout (#17)
- Issue when spending rules calculate wrong amount of max point to spend (#19)
- Added ProxiBlue GiftPromo detection to Validator
- Fixed Summary Block display at Customer Account Dashboard
- Fixed insecure link for Reward Block at Cart
- Fixed an issue with earning twitter points for new Twitter API

Features
- Added Rich Pins support for Pinterest
- Created script for exporting customers' balances to CSV

1.1.7

(2016-04-26) Features
- Added conditions Discount Amount and Coupon Used to the Cart Rule

Fixed
- Fixed issue with incorrect variable handling in WYSIWYG editor in Earning and Notification rules
- Fixed layout of Rewards Block for IDev Checkout
- Fixed bug which prevents tinyMCE from correct images and widgets upload in rules
- Fixed points earning if we manually change the order status

1.1.6

(2016-03-31) Improvements
- Add French locale files

Fixed
- Fixed issue which prevents social buttons from earning points (affects only 1.1.5)
- Enabled translation for backend translations and fixed email template selection

1.1.5

(2016-03-16) Improved
- Added compatibility patch with Intenso-based themes, which allows spend block to be shown on cart
- Allow to always use Referred Customer properties in Behaviour Rule
- Compatibility with IdevCheckout v. 4.5.5

Fixed
- Fixed referral record on registration from social network and fixed field with sharing URL

- 3rd-party extensions can change the quote id for order and this may be cause of error in backend
- Fixed referral record on registration from social network and fixed field with sharing URL
- Fixed behaviour event Customer logs in, now it earn points once a day
- Added script disable script for Referral condition
- Added missed translation strings to CSV files
- Fixed behaviour event Customer logs in, now it earn points once a day
- Fixed bug, which forced all notifications to be sent from General Contact email
- Fixed bug with not cancelling points, awarded for order from referral

1.1.4

1.1.3

Fixed
- Added validator test for known issues and check for RWP-109 added
- Fixed very old bug with not cancelling points, awarded for order from referral. Affects all versions
- RWP-107
- RWP-100
- RWP-106
- RWP-70
- RWP-71
- RWP-68

1.1.2

Fixed
- RWP-53

1.1.1

Fixed
- CSS fix for RWP-34
- RWP-43

1.0.0

Features
- Initial stable release