

koma·loom

Tech Integration Guide 1.5

Table of Contents

[Table of Contents](#)

[1 Scope of the document](#)

[1.1 What is kooma·looma](#)

[2 Terms Definition](#)

[3 Integration steps](#)

[3.1 How the integration works](#)

[3.2 Example Scenario](#)

[3.3 Create a kooma·looma Partner Account](#)

[3.4 Create your first campaign](#)

[3.5 Integrate kooma·looma on your web frontend](#)

[3.5.1 The kooma·looma javascript](#)

[3.5.2 The kooma·looma html widget “earn points”](#)

[3.5.3 The kooma·looma html widget for “thank you page”](#)

[3.6 Integrate kooma·looma on your backend](#)

[Identify the user you want to assign points to](#)

[Assign the points to the user](#)

[How to Test the Integration](#)

1 Scope of the document

This document explain you how to make the technical integration of kooma·looma web frontend widgets and kooma·looma backend API.

1.1 What is kooma·looma

kooma·looma is an Open Coalition Loyalty Program and Platform that allows E and M-Commerce merchants and App providers to easily manage promotions and reward programs on their stores, increase loyal users and attract new customers.

kooma·looma provides its partners a loyalty Platform as-a-Service with easy to integrate APIs and widgets designed for mobile/web browsers and the app ecosystem. The solution offers a web panel that allows partners to manage promotions, analyse aggregated customer behaviours and make data driven marketing decisions.

2 Terms Definition

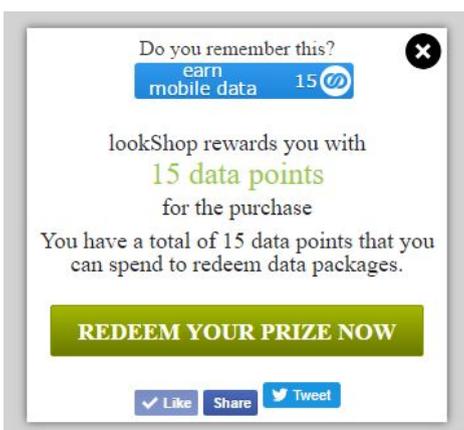
kooma·looma points: the basic measuring unit utilized in the kooma·looma programme. Points earned and accumulated by Members can be redeemed by them to obtain a Reward.

Funding rate: the ratio between the economical value of the points assigned when a customer purchases a product and the price of the product. This is a measure of the marketing investment done to incentivize the purchase of the product. Example: a funding rate of 5% means that for the purchase of a product priced 10.00\$ you assign to the customer an amount of points that worths 0.50\$.

kooma·looma “earn points” widget: a “badge” that show how many points a customer can earn when he completes a specific action on your web site (make a purchase, signup to a newsletter, invite a friend...). Here an example



kooma·looma “thank you” widget: an html fragment that show the current points balance of a user and invite him to enter the kooma·looma catalog site to redeem a reward. Here an example



kooma-looma javascript snippet: a javascript code that you should add in the html of the pages of your frontend web site that enables the rendering of the kooma-looma widgets.

Transactional Activity: an activity on your site associated to an economic value. Example: purchase of a product, download of a paid digital content.

Not Transactional Activity: any activity on your site that is not associated to an economic transaction. Examples: signup a newsletter, make a review of a product, invite a friend, fill up a detailed user profile.

Property: you might think a “property” as a distinct website or mobile app.

Campaign: consist of specific actions designed to incentivize both transactional and not transactional activities in a given Property. kooma-looma associates reporting and analytics data to each campaign.

ACQUISITION campaign: a campaign with a specific goal “increase my customer base”. Example: reward a customer with some loyalty points when he signup to your web site.

ENGAGE campaign: a campaign with a specific goal “more interactions from your users”. Example: reward a customer with some loyalty points when he write a review of your product.

RETENTION campaign: a campaign with a specific goal “keep users loyal to my company”. Example: reward a customer with some loyalty points when he purchase a product on your site

LIFT campaign: a campaign with a specific goal “increase average spending of users”. You divide your products in group A “low price products” and group B “high price products” and you assign more points when a user purchase a product from group B. In this way you can incentivize the purchase of higher priced products. Example: reward a customer with 4% founding rate when he purchase a product priced 20\$, but 6% for a product priced 30\$.

SHIFT campaign: a campaign with a specific goal “spend more on different products”. You divide your products in group A “low margin products” and group B “high margin products” and you assign more points when a user purchase a product from group B. In this way you can incentivize the purchase of higher margin products. Example: reward a customer with 4% founding rate when he purchase a product priced 20\$ whit a 20% margin for you, but 6% for a product priced 20\$ with 50% margin for you.

web frontend: a website that sells digital products or physical goods online.

kooma-looma backend API: a set of RESTful API that you can use to interact with the kooma-looma platform to assign loyalty points to a customer.

3 Integration steps

3.1 How the integration works

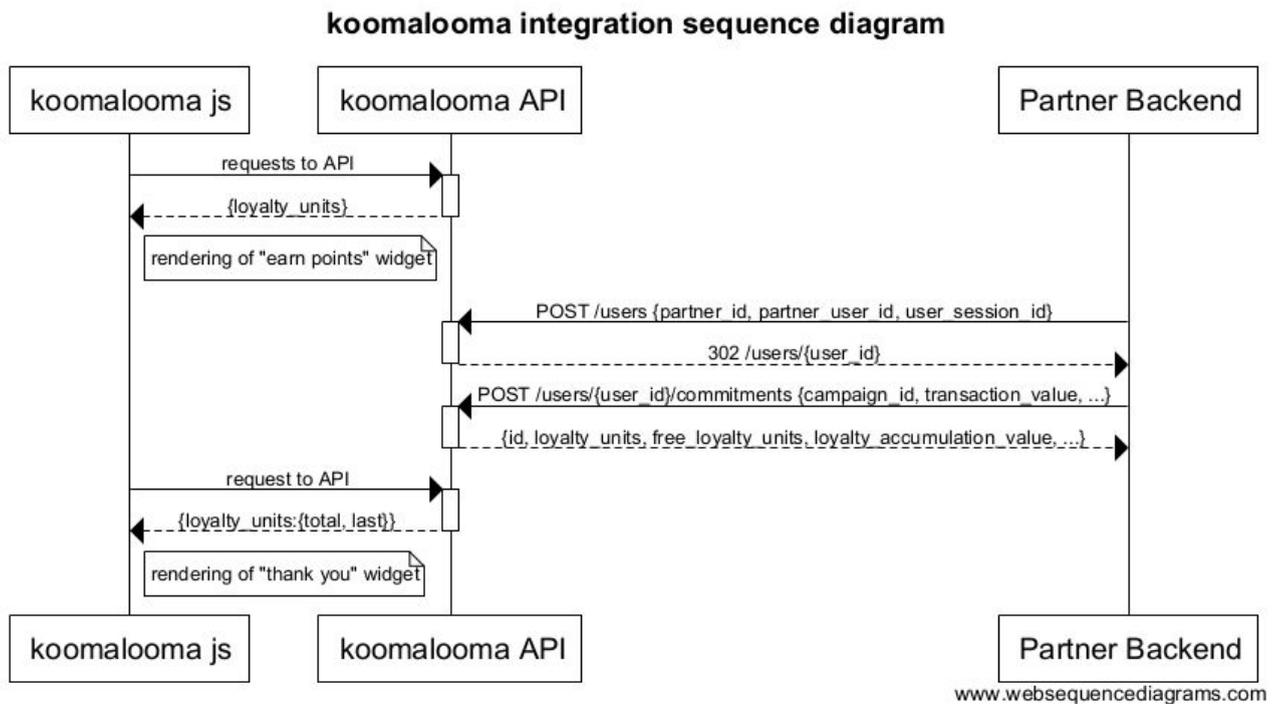
The integration of your web frontend with kooma-looma requires:

- The integration of the kooma-looma widgets and kooma-looma javascript in the html pages of your web frontend

- The integration of the kooma-looma backend API from the backend of the your web frontend.

The rendering in page of the kooma-looma widgets is done by the kooma-looma javascript.

Here is the sequence diagram that describe how the kooma-looma widgets, the kooma-looma javascript snippet, the kooma-looma backend API and the backend of your store collaborate to assign loyalty points to a specific customer.



We can identify 3 main phases in this sequence diagram:

- **Phase 1:** kooma-looma js interacts with kooma-looma backend API to show the kooma-looma “earn points” widget in the frontend html page
- **Phase 2:** the Partner backend interacts with kooma-looma backend API to effectively assign points to a specific user.
- **Phase 3:** kooma-looma js interacts with kooma-looma backend API to show the “thank you” widget in frontend html page

!!!IMPORTANT!!! All the interactions between actors “kooma-looma js” and “kooma-looma API” are handled by the kooma-looma javascript automatically, while you are in charge of the implementation of the 2 requests from actor “Partner Backend” to “kooma-looma API”, as described in the next chapters.

3.2 Example Scenario

To better understand how to integrate kooma-looma in your web frontend, let use a realistic example scenario and look at the required steps.

In our example scenario we assume that you have a web store named “lookshop” and that you want to incentivize your customers to spend more, giving “points” for each product purchased.

Let assume that “lookshop” has a structure like this:

- A web page “products.html” where some of the products are shown;

- A page “cart.html” that shows the items present in the customer’s cart and starts the payment process
- A page “thankyou.html” where the purchase made is confirmed, where the customer lands after the payment process.

In this scenario, we want to:

- Show the kooma·looma “earn points” widget to the page “product.html” to show how many points the customer can get for each different product
- Show the kooma·looma “earn points” widget to the page “cart.html” to show how many points the customer will get after the payment of the items in his cart.
- Show the kooma·looma “thank you” widget to the page “thankyou.html” to show the current points balance to the customer and suggest to see the kooma·looma catalog to redeem a prize.

In the next few chapters we’ll see how to implement this scenario.

3.3 Create a kooma·looma Partner Account

To integrate and use the kooma·looma platform you need a Partner Account. To create your account, just open the url <https://partner.kooma-looma.com/> in your browser and fill the signup form.

3.4 Create your first campaign

When the signup process completes, a “wizard” starts that guides you in the creation of your first campaign in 2 easy steps:

1. **Create a property:** for our example scenario we create a property named “lookshop web site”
2. **Add a campaign to the property:** for our example scenario we create a RETENTION campaign named “incentivize purchase” and we assign 3% of the value of the product in points.

3.5 Integrate kooma·looma on your web frontend

At the end of the step 2 “add a campaign to property” you’ll see on the screen the required html code and javascript that you should add to your frontend store. Let’s go in the details.

3.5.1 The kooma·looma javascript

This is the javascript that “does the magic”: you should add this in all the pages where a kooma·looma widget is shown.

In our scenario we should add this javascript “products.html”, in the “cart.html” and in the “thankyou.html”

!!!IMPORTANT!!! You should change the parameter %%sessionId%% with a unique string for each user session.

```
<script>
    (function(d, s, id) {
        var js, kljs = d.getElementsByTagName(s)[0];
        if (d.getElementById(id)) return;
        js = d.createElement(s); js.id = id;
        js.src = "//api.kooma-looma.com/sdk/js/sdk.js";
        kl = {userSessionId: %%sessionId%%};
        kljs.parentNode.insertBefore(js, kljs);
    })
```

```
        }(document, 'script', 'kl-sdk'));  
</script>
```

3.5.2 The kooma·looma html widget “earn points”

Here is the html code that you should add in your page where you want to show the kooma·looma widget “earn points”.

```
<div class="kl-widget-container" data-action="loyalty-units-for"  
data-campaign-id="573c824d22b5287393119ef1"  
data-transaction-value="{TRANSACTION_VALUE}" ></div>
```

!!!IMPORTANT!!! The parameter data-campaign-id is auto generated when you create a campaign, you should not change it.

!!!IMPORTANT!!! You should change the parameter {TRANSACTION_VALUE} with the value of the transaction associated to the widget. The format of this parameter is “float/currency_symbol”

Examples: “120.20/USD”, “510.90/NGN”

You can have as multiple kooma·looma “earn points” widgets in the same page, as many as you want. This is useful when you want to show multiple products on the same page.

In our scenario we should:

- add that html code to the page “products.html” for each product, setting in “TRANSACTION_VALUE” the value of the product associated to each widget.
- add that html code to the page “cart.html” and use the total amount of the cart in “TRANSACTION_VALUE”.

3.5.3 The kooma·looma html widget for “thank you page”

Here is the html code that you should add in your page where you want to show the kooma·looma widget “thankyou”.

```
<div class="kl-widget-container" data-action="last-commitment"  
data-user-id="{USER_ID}" data-partner-id="573c80bb22b52804dd4d6e81" ></div>
```

!!!IMPORTANT!!! The parameter data-partner-id is auto generated when you create a campaign, you should not change it.

!!!IMPORTANT!!! You should change the parameter {USER_ID} with the user_id that you received from the request to kooma·looma backend API (see chapter 3.6)

3.6 Integrate kooma·looma on your backend

The integration on backend side requires just 2 request from you backend to kooma·looma Backend API, as depicted in the sequence diagram 3.1.

In our example scenario, you make that 2 calls to the kooma·looma API after the customer completed the payment process started in page “cart.html”, and before showing the page “thankyou.html”.

Here are the Requests required

3.6.1 Identify the user you want to assign points to

The aim of the first request is to identify the user that you want to assign points to.

To identify that specific user, you make an HTTP POST request with some specific Attributes passed as a json in the BODY, according to this schema

| REQUEST | |
|----------|--------|
| Method | POST |
| Endpoint | /users |

| HEADERS | |
|------------------|---|
| Content-Type | application/json |
| X-Koomalooma-Jwt | The JWT token assigned to you by your kooma·looma technical support specialist. Example: eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOiJ0UEd2UWZ3QnhLN1ZKdzJVRXp3aTBtZEI0RGR0eUtSRnZOX2hYdUN0cTlvLCIsImZlcyY6InRlc3QuYXBpLmtvb21hbG9vbWEuY29tliwiaWF0IjoxNDYzNjQ1NjE1LCJhdWQiOiJMYW5nLCBFbWFyZCBhbmQgTydLZWVmZSIsInBhcnRuZXJfaWQiOiI1NzN |

| ATTRIBUTES | |
|-----------------|--|
| partner_id | Your partner_id. You can use the value of data-partner-id parameter |
| partner_user_id | A unique string that identify your customer. !!IMPORTANT!! You should use always the same partner_user_id for the same customer |
| user_session_id | A unique string that identify the current user session. !!IMPORTANT!! This value should be the same set in %%sessionId%% of kooma·looma javascript (see 3.5.1) |

Example of Request json:

```
{
  "partner_id": "573d75afe779890bd44d7849",
  "partner_user_id": "Bob-acme",
  "user_session_id": "Bob\\/51d4bfea-317c-4a44-b7e3-1e82c54d70c4"
}
```

In the HTTP Response you get a json representation of the User resource.

Example of Response Body:

```
{
  "id": "573d75aee779890bd44d783d"
}
```

3.6.2 Assign the points to the user

You can now assign points to the user identified by the “id” returned by the previous request.

We call this request a “Commitment”, because it is a "promise" to buy points from kooma·looma, done by an Partner, when a user completes a purchase.

You make an HTTP POST request with some specific Attributes passed as a json in the BODY, according to different schemas, depending on the type of campaign created.

Retention campaign

Here is the schema to use for a Retention campaign.

| REQUEST | |
|----------|--|
| Method | POST |
| Endpoint | /users/{id}/commitments Use the id received in the previous request as {id} |

| HEADERS | |
|------------------|--|
| Content-Type | application/json |
| X-Koomalooma-Jwt | The JWT token assigned to you by your kooma·looma technical support specialist. Example: eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOiJ0UEd2UWZ3QnhLN1ZKdzJVRXp3aTBtZEI0RGR0eUtSRnZOX2hYdUN0cTlvLCIsImZcyI6InRlc3QuYXBpLmtvb21hbG9vbWEuY29tliwiaWF0IjoxNDYzNjQ1NjE1LCJhdWQiOiJMYW5nLCBFbWFiZCBhbmQgTydLZWVmZSIsInBhcnRuZXJfaWQiOiI1NzN |

| ATTRIBUTES | |
|-------------|---|
| campaign_id | The campaign_id that this commitment refers to. |

| | |
|---|---|
| | !!!IMPORTANT!! You should use the value of parameter <code>data-campaign-id</code> of kooma·looma “earn points” widget, described in 3.5.2 |
| <code>transaction_value</code> | The price of the product purchased. The format is “float/currency_symbol” Examples: “120.20/USD”, “510.90/NGN” |
| <code>promotion_goal_description</code> | A string that describes the product purchased by the customer. Example: “shoes of brand xxx”. |
| <code>promotion_goal_category</code> | A string that describes the category of the product purchased by the customer. Example: “shoes and accessories”. |

Example of Request json:

```
{
  "campaign_id": "573d75ade779890bd44d781c",
  "transaction_value": "100.0000\\\/USD",
  "promotion_goal_description": "pair of shoes of brand xxx",
  "promotion_goal_category": "shoes and accessories"
}
```

In the HTTP Response you get a json representation of the Commitment resource.

```
{
  "id": "573d75ade779890bd44d781f",
  "loyalty_units": 500,
  "free_loyalty_units": 250,
  "loyalty_accumulation_value": "5%",
  "user_id": "573d75aee779890bd44d783d",
  "partner_id": "573d75afe779890bd44d7849",
  "goal": {
    "type": "retention",
    "value": "5%"
  },
  "redeemable_loyalty_units": 500,
  "redeemed_loyalty_units": 0,
}
```

```

"redeemed": false,
"confirmation_status": "confirmed",
"promotion_goal_description": "pair of shoes of brand xxx",
"promotion_goal_category": "shoes and accessories",
"created_at": "2016-01-19T08:13:33+00:00",
"updated_at": "2016-01-19T08:13:33+00:00"
}

```

Acquisition campaign

Here is the schema to use for an Acquisition campaign.

| REQUEST | |
|----------|--|
| Method | POST |
| Endpoint | /users/{id}/commitments Use the id received in the previous request as {id} |

| HEADERS | |
|------------------|--|
| Content-Type | application/json |
| X-Koomalooma-Jwt | The JWT token assigned to you by your kooma·looma technical support specialist. Example: eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOiJ0UEd2UWZ3QnhLN1ZKdzJVRXp3aTBtZEI0RGR0eUtSRnZOX2hYdUN0cTlvLCsImlzcyl6InRlc3QuYXBpLmtvb21hbG9vbWEuY29tliwiaWF0IjoxNDYzNjQ1NjE1LCJhdWQiOiJMYW5nLCBFbWFiZW50IiwiaWF0IjoiMj01NzN |

| ATTRIBUTES | |
|-------------|---|
| campaign_id | The campaign_id that this commitment refers to. !!!IMPORTANT!!! You should use the value of parameter <code>data-campaign-id</code> of kooma·looma “earn points” widget, described in 3.5.2 |

Example of Request json:

```
{
```

```

    "campaign_id": "57863a3ce7798947c05929ea"
  }

```

In the HTTP Response you get a json representation of the Commitment resource.

```

{
  "id": "57863a3ce7798947c05929ed",
  "loyalty_units": 50,
  "free_loyalty_units": 25,
  "loyalty_accumulation_value": "0.5000\ /EUR",
  "user_id": "57863a3ae7798947c05929d1",
  "partner_id": "57863a3be7798947c05929dd",
  "goal": {
    "type": "acquisition",
    "value": "0.5000\ /EUR"
  },
  "redeemable_loyalty_units": 50,
  "redeemed_loyalty_units": 0,
  "redeemed": false,
  "confirmation_status": "confirmed",
  "promotion_goal_description": "A new campaign",
  "promotion_goal_category": "acquisition",
  "created_at": "2016-07-13T12:55:24+00:00",
  "updated_at": "2016-07-13T12:55:24+00:00"
}

```

Engage campaign

Here is the schema to use for an Engage campaign.

| | |
|----------------|--|
| REQUEST | |
| Method | POST |
| Endpoint | /users/{id}/commitments Use the id received in the previous request as {id} |

| HEADERS | |
|------------------|---|
| Content-Type | application/json |
| X-Koomalooma-Jwt | The JWT token assigned to you by your kooma-looma technical support specialist. Example: eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOiJ0UEd2UWZ3QnhLN1ZKdzJVRXp3aTBtZEI0RGR0eUtSRnZOX2hYdUN0cTlvLCIsImVzcyI6InRlc3QuYXBpLmtvb21hbG9vbWEuY29tliwiaWF0IjoxNDYzNjQ1NjE1LCJhdWQiOiJMYW5nLCBFbWFiZCBhbmQgTydLZWVmZSIsInBhcnRuZXJfaWQiOiI1NzN |

| ATTRIBUTES | |
|-------------|---|
| campaign_id | The campaign_id that this commitment refers to. !!!IMPORTANT!!! You should use the value of parameter <code>data-campaign-id</code> of kooma-looma “earn points” widget, described in 3.5.2 |

Example of Request json:

```
{
  "campaign_id": "57863b4ee77989485c67a49a"
}
```

In the HTTP Response you get a json representation of the Commitment resource.

```
{
  "id": "57863b4ee77989485c67a49d",
  "loyalty_units": 50,
  "free_loyalty_units": 25,
  "loyalty_accumulation_value": "0.5000\ /EUR",
  "user_id": "57863b4ce77989485c67a481",
  "partner_id": "57863b4de77989485c67a48d",
  "goal": {
    "type": "engage",
    "value": "0.5000\ /EUR"
  },
  "redeemable_loyalty_units": 50,
```

```
"redeemed_loyalty_units":0,  
"redeemed":false,  
"confirmation_status":"confirmed",  
"promotion_goal_description":"A new campaign",  
"promotion_goal_category":"engage",  
"created_at":"2016-07-13T12:59:58+00:00",  
"updated_at":"2016-07-13T12:59:58+00:00"  
}
```

4 How to Test the Integration

kooma-looma provides a test environment where you can create as many Properties and Campaigns as you want, and develop and test your integration code freely.

Here are the urls to use.

- Partner site: <https://partner-ci.koomalooma.com> , signup here for a test account
- kooma-looma backend API base url: <https://api-ci.koomalooma.com>