Payment

Payment is a new feature in [Tiki5](#)

The Payment API is built to be hooked from multiple parts of the system. While there is a user interface, the primary purpose of it is for management. Payment requests should be hooked to actions in the system. A good example is the [PluginMemberPayment](#).

A payment request is like an [Invoice](#).

A registered payment will execute a set of behaviors when events occur. Supported events are 'complete', for when the payment is fully paid, and 'cancel' when the payment is abandoned. The behaviors can only be registered through the API. Manually created payments must be handled manually.

Behaviors are stored in **lib/payment/behavior/** and are named like the behavior called ('sample' would be in sample.php). Inside those files, a function named payment_behavior_sample is expected to be present. When the behavior is registered, an array is provided.

Initialize a payment

```
<?php global $paymentlib; require_once 'lib/payment/paymentlib.php'; $user = 'test123'; // ... $invoice = $paymentlib->request_payment( 'Some payment', 100, 30 ); $paymentlib->register_behavior( $invoice, 'complete', 'notify_admin', array( $user, 'All good!' ) ); $paymentlib->register_behavior( $invoice, 'cancel', 'notify_admin', array( $user, 'Forget it.' ) ); ?>
```

Upon payment, the following call will be made indirectly:

```
payment_behavior_notify_admin( 'test123', 'All good!' );
```

Wishes

For Shopping Cart and Payments.

Open & Pending

{*WISHES(filtervalue="234", status="op")*}{*WISHES*}

Closed

[+]

Extension

In themselves, the payments only handle the actual request for money and tracking of the payment made, regardless of the payment technique used. In a full payment process, you may have something that looks like this:

1. Product selection
2. Applying discount
3. Calculate shipping charges
4. Calculate applicable taxes
5. Proceed to payment (register the payment, as we care about it here)
6. Perform payment from gateway

Currently, the product selection can be handled through the [cart module](#) and [add to cart plugin](#). Other

elements in-between are not implemented at this time. When extending, please keep in mind the entire process and avoid adding too many features to the payment handling directly.

Team

- MarcLaporte
- Louis-Philippe Huberdeau
- You?

Paypal

https://developer.paypal.com/ ⬈

To help with development, here you are create a sandbox account

Related

- Payments Pro
- PayPal Subscription
- Refunds
- https://www.dwolla.com ⬈
- Bitcoin
- Community Currencies
- https://github.com/stripe/jquery.payment ⬈
- http://omnipay.thephpleague.com/ ⬈
- http://money.readthedocs.org/en/latest/Introduction.html ⬈
- https://github.com/ziadoz/awesome-php/blob/master/README.md#e-commerce ⬈

Alias names of this page:
Payments