

به نام خدا

مستندات مازول Node.js فراگیت

نویسنده: عرفان صحاف تژاد

مرداد 95

توضیحات

این ماژول جهت استفاده آسان از وب سرویس و API های درگاه واسط پرداخت فراگیت در پلتفرم Node.js نوشته شده و بر روی بستر مدیریت پکیج NPM منتشر شده است. جهت مشاهده مستندات به زبان انگلیسی میتوانید به [این مخزن گیت هاب](#) مراجعه کنید. جهت درک بهتر مقادیر دریافتی وب سرویس و پارامتر های اختیاری پیشنهاد میشود یکبار [مستندات REST-API](#) موجود در آزمایشگاه فراگیت را به علاوه مستندات انگلیسی همین پکیج را مطالعه فرمایید.

ما در این پکیج 3 متد اصلی کار با وب سرویس را پیاده سازی کردیم که عبارتند از: درخواست پرداخت – تایید پرداخت – دریافت موجودی اکانت جهت نصب پکیج مربوطه میبایست از دستور زیر استفاده نمایید:

```
npm install faragate --save
```

حال این ماژول آماده تعریف در سورس کد و استفاده است.

متد paymentRequest

برای هدایت کاربر به صفحه درگاه های پرداخت میبایست ابتدا از این متد استفاده کنید. این متد دارای 5 پارامتر ضروری و 1 پارامتر اختیاری با نوع Object است.

ورودی اول همان res موجود در Route مربوطه است که مستقیماً باید به این متد داده شود. ورودی دوم مقدار یکتا مرچنت درگاه است که در پنل کاربری شما موجود است. ورودی سوم مبلغ تراکنش بصورت ریال است. ورودی چهارم آدرس برگشتی از درگاه بصورت رشته است. ورودی پنجم شماره فاکتور است که میبایست بصورت یکتا آن را تولید کنید و به این متد دهید. ورودی ششم نیز همان پارامتر های

اختیاری تحت قالب Object بصورت Key:Value است مانند نام پرداخت کنند، ایمیل، نام مازول، اکسترا مرچنت و ...

نمونه کد استفاده از این متد در فریمورک اکسپرس (در Production):

```
var faragate = require("faragate");
var marchent = "YOUR_MARCHENT_CODE";

app.get('/pay', function(req, res){
  faragate.paymentRequest(res, marchent, "RIAL_PRICE",
    "http://YOUR_WEBSITE.com/pay", "INVOICE_NUMBER", {PaymentName: "Erfan Sahafnejad"});
});
```

توجه: هیچگاه در Route مربوط به انتقال کاربر به درگاه پرداخت، از متد های res مانند res.send و res.end استفاده نکنید زیرا مانع ریدایرکت کاربر خواهد شد.

متد verifyPayment

بعد از هدایت کاربر به سمت درگاه (چه در صورت کنسل شدن پرداخت و چه در صورت موفقیت آمیز بودن آن)، کاربر به آدرس کال بک نوشته شده در متد قبلی فرستاده خواهد شد. حال زمان بررسی وضعیت پرداخت است که Route مورد نظر میبایست بر روی متد POST تنظیم شده باشد. در این حالت، درگاه پرداخت فراگیت مقادیر توکن، شماره فاکتور و وضعیت سند باکس (در صورت True بودن در متد قبلی) برای ما ارسال خواهد کرد.

این متد دارای 3 ورودی ضروری و 1 ورودی اختیاری (سند باکس) است. ورودی اول مرچنت کد درگاه است. ورودی دوم توکن ارسال شده از سمت فراگیت است. ورودی سوم تابع Async شما جهت بررسی وضعیت پرداخت است. ورودی چهارم حالت سند باکس (Boolean) است، درصورتی که مقدار دهی نشود False در نظر گرفته میشود. (برای اطلاعات بیشتر راجع به SandBox به مستندات اصلی فراگیت مراجعه کنید.)

نمونه کد استفاده از این متد در فریمورک اکسپرس (در Production) :

```
var faragate = require("faragate");
var marchent = "YOUR_MARCHENT_CODE";

app.get('/', function(req, res){
  faragate.paymentRequest(res, marchent, "10000",
    "http://YOUR_WEBSITE.com/pay", "0123456789", {PaymenterName: "Erfan Sahafnejad"});
});

app.post("/pay", function(req, res){
  var token = req.body.Token;
  faragate.verifyPayment(marchent, token, function(status, code, message){
    var text = "Payment status: " + status + "<br>Code: " + code +
    "<br>Message: " + message;
    res.send(text);
  });
});
```

در ورودی سوم که تابع Async شما جهت بررسی وضعیت پرداخت است، 3 متغیر که نشانگر وضعیت، کد و توضیحات میباشد در دسترس است. Status بصورت Boolean بوده که در صورت موفقیت آمیز بودن تراکنش بصورت True خواهد بود. محتوای متغیر های Code و Message در مستندات فراگیت بصورت کامل توضیح داده شده.

متد getBalance

با استفاده از این متد میتوانید مقدار موجودی اکانت خود را به ریال دریافت کنید. برای استفاده از این متد میبایست رمز وب سرویس را در بخش تغییر رمز عبور اکانت خود تعیین و دریافت کنید.

این متد دارای 3 ورودی اجباریست و ورودی اختیاری نیز نخواهد داشت:

ورودی اول نام کاربری شما در فراگیت است. ورودی دوم رمز و پسورد وبسرویس اکانت شماست و ورودی سوم تابع Async جهت دریافت و استفاده از موجودی اکانت است.

نمونه کد استفاده از این متد در فریمورک اکسپرس (در Production) :

```
var faragate = require("faragate");
var marchent = "YOUR_MARCHENT_CODE";

app.get('/balance', function(req, res){
  faragate.getBalance("USERNAME", "PASSWORD", function(balance, code,
message){
    res.send(balance + " Rial" + "<br>Message: " + message);
  });
});
```

در تابع Async شما، 3 متغیر Balance با مقدار موجودی اکانت شما به همراه Code و Message در دسترس است که در مستندات RESTful فراگیت بصورت کامل توضیح داده شده است.

مشکلات و issue ها

در صورت وجود مشکل در پکیج میتوانید آن را در بخش issue ریپازیتوری گیت هاب با ما در میان گذاشته و به بهبود این ماژول و داشتن تجربه کاری بهتر دیگر توسعه دهندگان کمک کنید.

لایسنس و مجوز استفاده

این پکیج تحت لایسنس نرم افزار های متن باز Apache نسخه 2.0 میباشد.

به امید موفقیت تمامی توسعه دهندگان ایرانی! :

با تشکر، عرفان صحاف تژاد – فراگیت