Integración de Mercadopago

Nicolás Castro Garcia

@ncastrogarcia

- Frontend Engineer @ exactly.finance
- CTO @ CourseIt
- Fullstack web developer @ digiventures

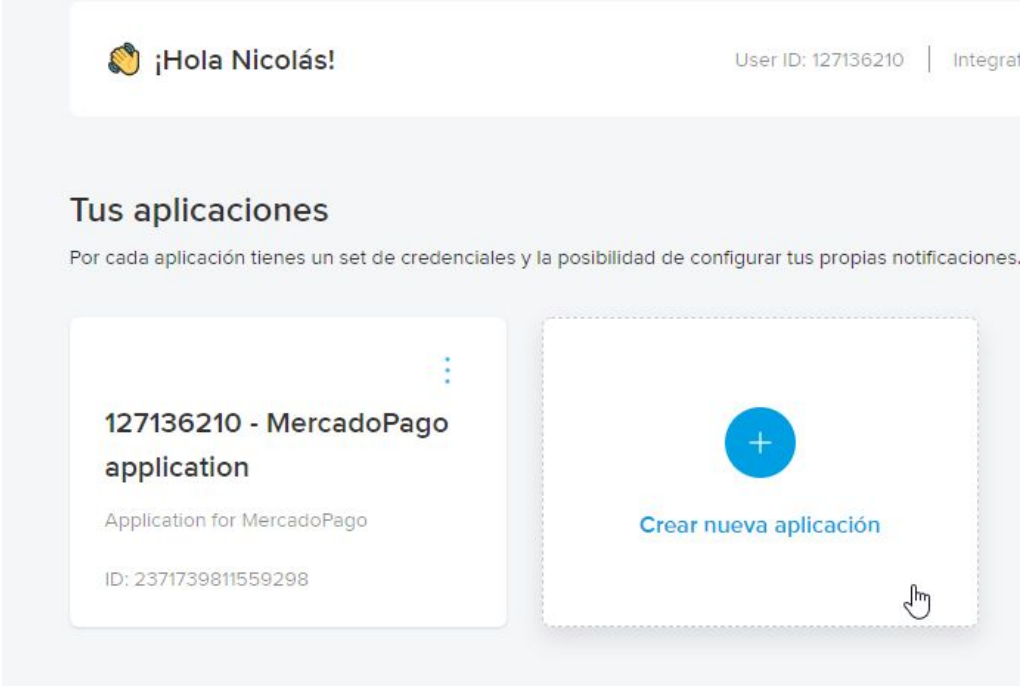
Requisitos

- Conocimientos de Express / javascript
- Manejo de terminal
- Api keys de mercadopago

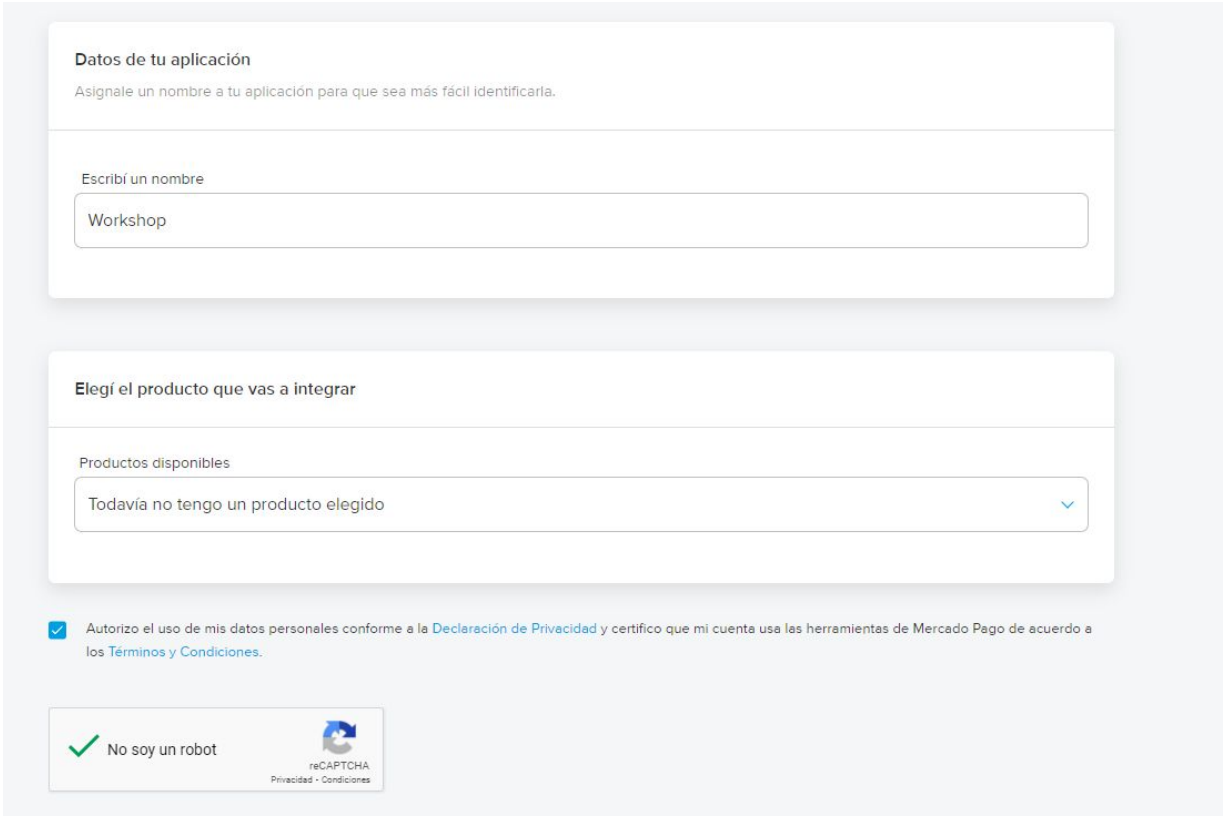
API KEYS

API Keys

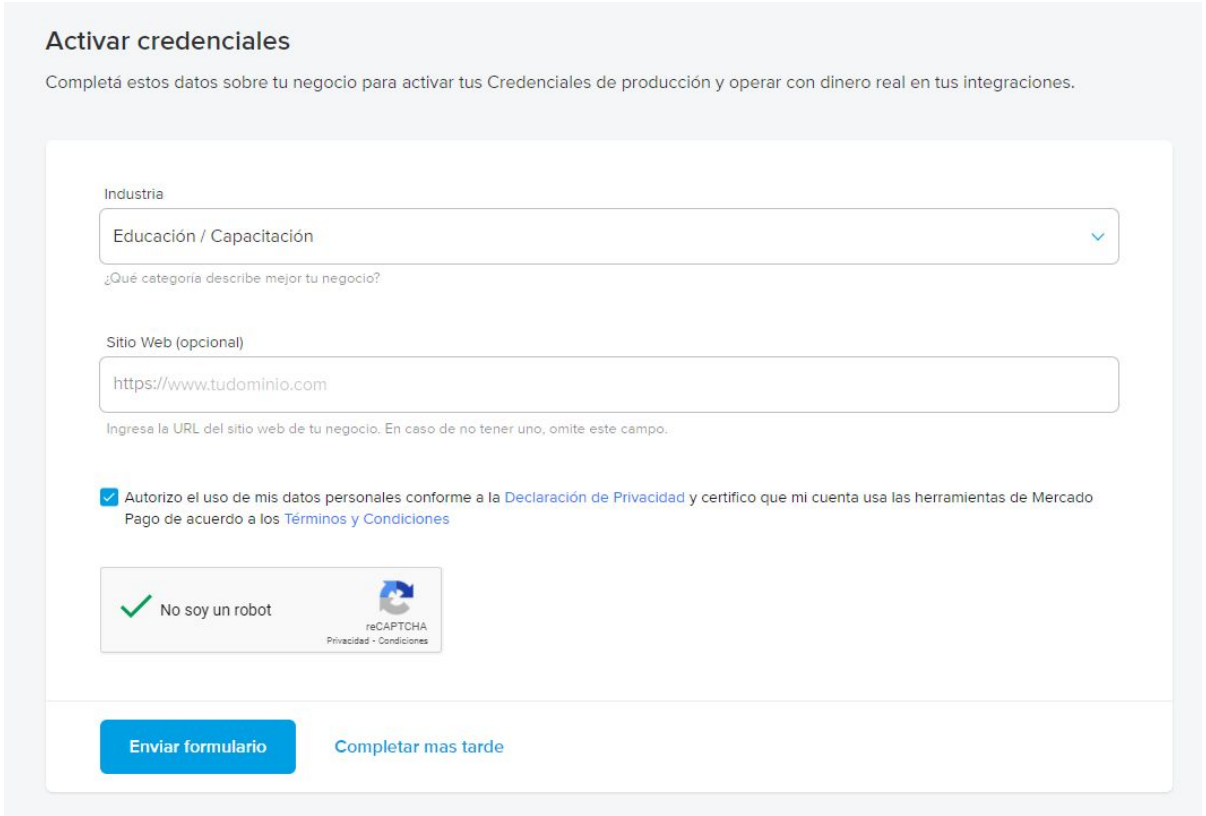
https://www.mercadopago.com.ar/developers/panel



API Keys

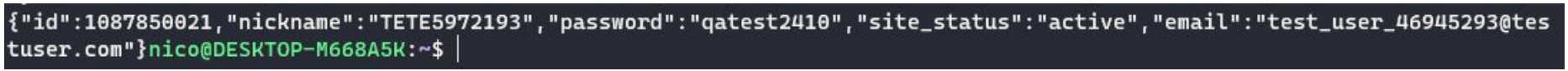

Datos de tu aplicación
Asignale un nombre a tu aplicación para que sea más fácil identificarla.
Escribí un nombre
Workshop
Elegí el producto que vas a integrar
Productos disponibles
Todavía no tengo un producto elegido
Autorizo el uso de mis datos personales conforme a la Declaración de Privacidad y certifico que mi cuenta usa las herramientas de Mercado Pago de acuerdo a los Términos y Condiciones.
No soy un robot
reCAPTCHA
Privacidad - Condiciones

API Keys


Activar credenciales
Completá estos datos sobre tu negocio para activar tus Credenciales de producción y operar con dinero real en tus integraciones.
Industria
Educación / Capacitación
¿Qué categoría describe mejor tu negocio?
Sitio Web (opcional)
https://www.tudominio.com
Ingresa la URL del sitio web de tu negocio. En caso de no tener uno, omite este campo.
Autorizo el uso de mis datos personales conforme a la Declaración de Privacidad y certifico que mi cuenta usa las herramientas de Mercado Pago de acuerdo a los Términos y Condiciones
No soy un robot
reCAPTCHA
Privacidad - Condiciones
Enviar formulario
Completar mas tarde

Usuarios de Prueba

```
curl -X POST \
-H "Content-Type: application/json" \
-H 'Authorization: Bearer PROD_ACCESS_TOKEN' \
"https://api.mercadopago.com/users/test_user" \
-d '{"site_id":"MLA"}'
```



Fuente: https://www.mercadopago.com.ar/developers/es/guides/online-payments/checkout-pro/test-integration

Configuración del Proyecto

Creación

- Ejecutamos el comando:
  - npx express-generator –view=none

- Ejecutamos npm install en la carpeta del proyecto

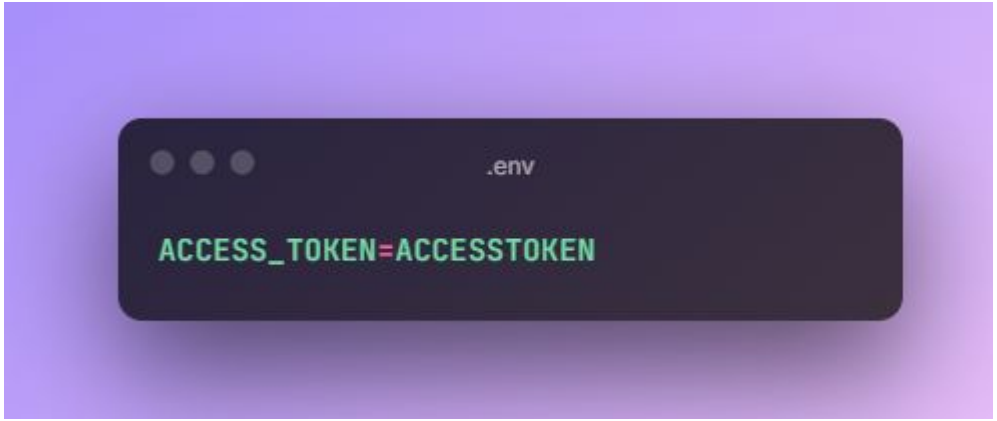
- Ejecutamos
  - npm install axios
  - npm install dotenv

DOTENV

app.js

```
const express = require("express");
const path = require("path");
const cookieParser = require("cookie-parser");
const logger = require("morgan");
const dotenv = require("dotenv");

const indexRouter = require("./routes/index");

const app = express();

dotenv.config();

app.use(logger("dev"));
app.use(express.json());
app.use(express.urlencoded({ extended: false }));
app.use(cookieParser());
app.use(express.static(path.join(__dirname, "public")));

app.use("/", indexRouter);

module.exports = app;
```

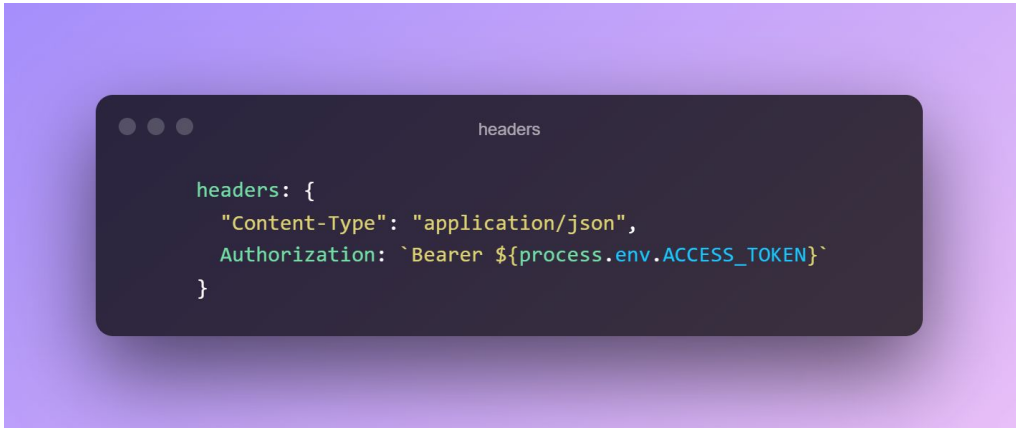
DOTENV



Configuración del Proyecto: Requests

Pagos Únicos

- https://api.mercadopago.com/checkout/preferences



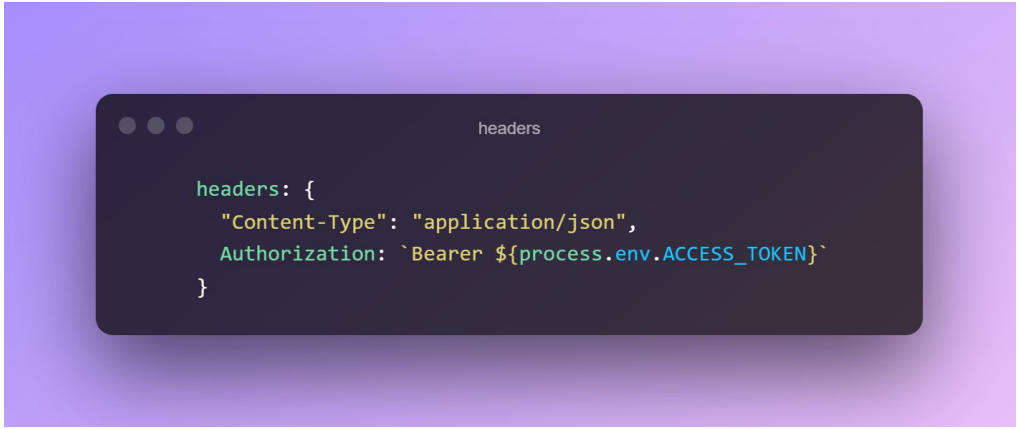
Fuente: https://www.mercadopago.com.ar/developers/es/guides/online-payments/checkout-pro/configurations

Pagos Únicos

Fuente: https://www.mercadopago.com.ar/developers/es/guides/online-payments/checkout-pro/configurations

Pagos Recurrentes

- https://api.mercadopago.com/preapproval



Fuente: https://www.mercadopago.com.ar/developers/es/reference/subscriptions/_preapproval/post

Pagos Recurrentes

Fuente: https://www.mercadopago.com.ar/developers/es/guides/online-payments/checkout-pro/configurations

Configuración del Proyecto: Rutas

Rutas

./routes/index.js

```
const express = require("express");
const router = express.Router();

const PaymentController = require("../controllers/PaymentController");
const PaymentService = require("../services/PaymentServices");

const PaymentInstance = new PaymentController(new PaymentService());

router.get("/", function (req, res, next) {
  return res.json({
    "/payment": "generates a payment link",
    "/subscription": "generates a subscription link"
  });
});

router.get("/payment", function (req, res, next) {
  PaymentInstance.getPaymentLink(req, res);
});

router.get("/subscription", function (req, res, next) {
  PaymentInstance.getSubscriptionLink(req, res);
});

module.exports = router;
```

PaymentController.js

```
class PaymentController {
  constructor(subscriptionService) {
    this.subscriptionService = subscriptionService;
  }

  async getPaymentLink(req, res) {
    try {
      const payment = await this.subscriptionService.createPayment();

      return res.json(payment);
    } catch (error) {
      console.log(error);

      return res
        .status(500)
        .json({ error: true, msg: "Failed to create payment" });
    }
  }

  async getSubscriptionLink(req, res) {
    try {
      const subscription = await this.subscriptionService.createSubscription();

      return res.json(subscription);
    } catch (error) {
      console.log(error);

      return res
        .status(500)
        .json({ error: true, msg: "Failed to create subscription" });
    }
  }
}

module.exports = PaymentController;
```

PaymentService.js

```
const axios = require("axios");

class PaymentService {
  async createPayment() {
    const url = "https://api.mercadopago.com/checkout/preferences";

    const body = {
      payer_email: "comprador@email.com",
      items: [
        {
          title: "Dummy Title",
          description: "Dummy description",
          picture_url: "http://www.myapp.com/myimage.jpg",
          category_id: "cat123",
          quantity: 1,
          unit_price: 10
        }
      ],
      back_urls: {
        success: "https://www.success.com",
        failure: "http://www.failure.com",
        pending: "http://www.pending.com"
      },
      notification_url: "https://www.your-site.com/ipn"
    };

    const payment = await axios.post(url, body, {
      headers: {
        "Content-Type": "application/json",
        Authorization: `Bearer ${process.env.ACCESS_TOKEN}`
      }
    });

    return payment.data;
  }

  async createSubscription() {
    const url = "https://api.mercadopago.com/preapproval";

    const body = {
      reason: "Suscripción de ejemplo",
      auto_recurring: {
        frequency: 1,
        frequency_type: "months",
        transaction_amount: 10,
        currency_id: "ARS"
      },
      back_url: "https://google.com.ar",
      payer_email: "comprador@gmail.com.ar"
    };

    const subscription = await axios.post(url, body, {
      headers: {
        "Content-Type": "application/json",
        Authorization: `Bearer ${process.env.ACCESS_TOKEN}`
      }
    });

    return subscription.data;
  }
}

module.exports = PaymentService;
```

GRACIAS

Links

- https://github.com/NicolasCastroGarcia/mercadopago-workshop
- https://www.postman.com/collections/f98a3e25efc8f634016f
- https://github.com/course-it/typescript-express-boilerplate
- @ncastrogarcia