JavaScript SDK Guide

Official SDK for Node.js and browser-based applications to interact with the Edgesense REST APIs.

Installation

For Node.js:

Install using npm:

npm install @edgesense/sdk

Or using yarn:

yarn add @edgesense/sdk

For Browser (via CDN):

<script src="https://cdn.edgesense.io/sdk/v1/edgesense.min.js"></script>

The SDK will be available globally as Edgesense.

Authentication

You will need an API key to initialize the SDK.
You can generate it from the Edgesense dashboard under:
Settings → API Keys → Generate Key

Node.js Example:

```
const { Edgesense } = require('@edgesense/sdk');


const client = new Edgesense({
  apiKey: 'YOUR_API_KEY'
});
```

Browser Example:

```
const client = new Edgesense({
  apiKey: 'YOUR_API_KEY'
});
```

Basic Usage

1. Get All Devices

```
client.devices.list().then(devices => {
  devices.forEach(device => {
    console.log(device.name, device.status);
  });
});
```

2. Register a New Device

```
client.devices.register({
  name: 'Motor-01',
  type: 'vibration',
  location: 'Assembly Line B'
}).then(device => {
  console.log('Registered:', device.id);
});
```

3. Send Telemetry Data

```
client.telemetry.send('DEVICE_ID', {
  temperature: 64.3,
  vibration: 0.02,
  timestamp: new Date().toISOString()
});
```

4. Fetch Telemetry History

```
client.telemetry.fetch('DEVICE_ID', {
  startTime: '2025-07-23T00:00:00Z',
  endTime: '2025-07-23T23:59:59Z'
}).then(data => {
  console.log(data);
});
```

SDK Structure

Edgesense(options)

Initialize SDK with the following options:

Option	Type	Required	Description
apiKey	string	✅	Your Edgesense API key
baseUrl	string	❌	Override API URL (optional)

Modules

devices

- list() – Get all devices
- get(deviceId) – Get a single device
- register(data) – Register new device
- update(deviceId, data) – Update device info
- delete(deviceId) – Delete device

telemetry

- send(deviceId, data) – Push telemetry
- fetch(deviceId, options) – Get telemetry history

Error Handling

Errors are returned as promises. Use .catch() or try/catch.

With Promises:

```
client.devices.list()
  .then(devices => console.log(devices))
  .catch(err => console.error('Error:', err.message));
```

With async/await:

```
try {
  const devices = await client.devices.list();
  console.log(devices);
} catch (e) {
  console.error('Error fetching devices:', e.message);
}
```

Testing (Node.js)

Use mocking libraries like nock or jest:

```
jest.mock('@edgesense/sdk');


test('should fetch devices', async () => {
  Edgesense.mockImplementation(() => ({
    devices: {
      list: jest.fn().mockResolvedValue([{ name: 'TestDevice' }])
    }
  }));


  const client = new Edgesense({ apiKey: 'test' });
```

```
  const devices = await client.devices.list();
  expect(devices[0].name).toBe('TestDevice');
});
```

Security Notes (Browser)

- Never expose admin API keys in browser environments.
- Use read-only or limited-scope keys when accessing from frontend.
- For secure actions (write/update), use a backend proxy.

Advanced: Update Configuration

```
client.devices.updateConfig('DEVICE_ID', {
  sampling_rate: '30s',
  alertThresholds: {
    pressure: 100
  }
});
```

References

- REST API Docs
- Python SDK Guide
- System Requirements & Installation Manual
- Dashboard Walkthrough

Contact us for any query at :
Email : support@edgesense.io