

Nimmsta Library Documentation

Version 1.0.0

Since this document is just an example of SDK documentation. For simplicity, certain sections have been omitted and are represented by red placeholders, indicating their intended content instead of providing detailed information.

Date	Version	Update Log
21.1.2024	1.0.0	Initial version

Table of Contents

1. Introduction
2. Getting Started
3. Usage
 - 3.1. Initialization and configuration
 - 3.2. Handling User data
 - 3.3. Error Handling
4. Support and Contact
5. License

1. Introduction

This document provides documentation for the Nimmsta Library, a Dart package designed for seamless interaction with RESTful APIs provided by **{JSON} Placeholder**. It simplifies common API tasks such as fetching, creating, updating, and deleting data, offering a user-friendly interface for API interactions.

2. Getting Started

<Similar to steps stated in nimmsta_lib package ReadMe>

3. Usage

3.1 Initialization and Configuration

Nimmsta library doesn't need explicit initialization. It lazily self-initializes infrastructure as needed when relevant code gets executed. When no explicit initialization is run, the library will fall back to default configurations, as below.

<Table with default configuration values>

On the other hand, if custom configuration is needed, an explicit initialization with these custom configurations could be executed. To do this place the following code at the entry point of the application.

```
NimmstaLib.initialize(  
  config: NimmstaConfig(  
    timeout: Duration(  
      seconds: 15,  
    ),  
  ),  
));
```

The method **initialize** initializes the library with a custom **NimmstaConfig** object containing the custom configuration.

3.2 Handling user data

UserClient class acts as a gateway for all operations related to managing User data. It can be instantiated multiple times across various scopes within the application without performance implications. It gets all its infrastructure configurations from the configurations explained in point 3.1.

<Examples on how to use each of the operations UserClient, as shown in the package ReadMe>

3.3 Error handling

<Exception types, and when each type of exception is thrown by the library>

4. Support and Contact

For questions and feedback, please contact support@nimmsta.com.

5. License

<License information>