PRXClient IOS Integration

|  |  |  |  |
| --- | --- | --- | --- |
| Document History | | | |
| Version | Date | Section | Changes |
| 1.0.0 |  | All | Initial Release |
| 2.0.0 | 11- Jan-2017 | All | Now using Appinfra |
| 2.4.0 | 7-JUL-2017 | All | Added parent tla |

|  |  |
| --- | --- |
| Author | Ravi Kiran HR, Hashim MH, Leslie Sebastian |
| Approved by |  |
| Email ID | [leslie.sebastian@philips.com](mailto:leslie.sebastian@philips.com)  [Ravi.kiran@philips.com](mailto:Ravi.kiran@philips.com)  [hashim.mh@philips.com](mailto:hashim.mh@philips.com) |

Table of Contents

[Introduction. 3](#_Toc471832963)

[1 Client 3](#_Toc471832964)

[2 RequestManager 3](#_Toc471832965)

[3 Responsehandler 3](#_Toc471832966)

[4 Product/ProductSummary/ProductAssets 3](#_Toc471832967)

[Installation 3](#_Toc471832968)

[Code Example - Quick integration 3](#_Toc471832969)

[Create request 3](#_Toc471832970)

[Execute Request 3](#_Toc471832971)

# Introduction.

The main functionality of this library is to download any data related to product present on PRX. It can be used by consumer care, registration and different applications. This library can be reused by other projects with minimal development changes as a generic network component as well. PRX client library exposes classes and APIs to clients to send a request and get a response. Library also helps clients to customise the requests.

1 Client

It can be an application, consumer care component or registration component.

2 Request Manager

It provides set of public APIs for placing requests from client and also talks to Network wrapper class for performing network operations.

3 Response handler

Handles response like invoking respective builders to build the response. It also invokes listener/blocks.

4 Product/ ProductSummary/ ProductAssets Model data for each request type.

# Installation

* Using cocoapods - add Philip’s cocoapod repo and add PRXClient to podfile

pod 'PhilipsPRXClient'

* Using Frameworks -

# Code Example - Quick integration

#import "PRXRequestManager.h"

#import "PRXAssetRequest.h"

### 

### Create request

### 

PRXAssetRequest \*data=[[PRXAssetRequest alloc] initWithSector:B2C ctnNumber:@"HP8632/00" catalog:CONSUMER];

### Execute Request

Create Request Manager with appinfra and TLA call execute API

Appinfra -> instance of application appinfra

TLA -> three letter acronym of the calling common component

PRXDependencies \*dependencies = PRXDependencies \*dependencies = [[PRXDependencies alloc]initWithAppInfra:appInfra parentTLA:@"dcc"];

PRXRequestManager \*requestManager = [[PRXRequestManager alloc]initWithDependencies:dependencies];

[requestManager execute:data completion:^(PRXResponseData \*response)

{

}failure:^(NSError \*error)

{

}];