|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document History** | | | | | | |
| **Version** | | **Date** | **Author** | **Section** | **Changes** |
| 1.0 | | 10-07-2015 | Deepthi Shivakumar | All | Release 2 features |
| 1.1 | | 30-09-2015 | Sameer Sulaiman | Integration section | Bazaar voice feature changes. |
| 1.2 | | 1-10-2015 | Deepthi Shivakumar | Integration section | Bazaar voice feature changes. Added API details. |
| 1.3 | | 12/12/2015 | Sameer Sulaiman | Integration section | Modified changes for 15.2 PI release. |
| 1.4 | | 13/05/2016 | Sameer Sulaiman | Integration section | Modified changes for 16.2 PI release |
| 1.5 | | 22/07/2016 | Sameer Sulaiman | Integration section | Modified changes for 16.3 PI release |
| 1.6 | 8/09/2016 | Sameer Sulaiman | Integration section | Modified changes for 16.4 PI release |
| 1.7 | 1/12/2016 | Sameer Sulaiman | Integration section | Modified changes for 16.5 PI release |
| 1.8 | 09/02/2017 | Sameer Sulaiman, Niharika Bundela | Integration section | Modified changes for 17.1 PI release |
| 1.9 | 20/03/2017 | Sameer Sulaiman | Integration section | Modified changes for 17.2.2 PI release |
| 2.0 | 27/04/2017 | Sameer Sulaiman | Integration section | Modified changes for 17.2 PI release |
| 2.1 | 06/07/2017 | Sameer Sulaiman | Integration section | Modified changes for 17.3 PI release |
| 2.2 | 13/09/2017 | Sameer Sulaiman | Integration section | Modified changes for 17.4 PI release |

|  |  |
| --- | --- |
| **Author** | Deepthi Shivakumar, Sameer Sulaiman, Niharika Bundela |
| **Approved by** | Deepthi Shivakumar |
| **Email Id** | [Deepthi.Shivakumar@philips.com](mailto:Deepthi.Shivakumar@philips.com) [Sameer.Sulaiman@philips.com](mailto:Sameer.Sulaiman@philips.com)  [Niharika.Bundela@philips.com](mailto:Niharika.Bundela@philips.com) |

Table of Contents

[1. INTRODUCTION 3](#_Toc482183483)

[2. INTEGRATION 3](#_Toc482183484)

[2.1 Cocoa pod Integration 3](#_Toc482183485)

[2.2 Library Integration 3](#_Toc482183486)

[2.3 Prerequisites 4](#_Toc482183487)

[3. INITIALIZATION 4](#_Toc482183488)

[3.1 Consumer Care Configuration 7](#_Toc482183489)

[*3.1.1* Main menu configuration 9](#_Toc482183490)

[*3.1.2* Product menu configuration 10](#_Toc482183491)

[*3.1.3* Social provider configuration 11](#_Toc482183492)

[4. Vertical features customization 12](#_Toc482183493)

[5. Navigation bar customization 13](#_Toc482183494)

[6. Tagging 14](#_Toc482183495)

**[7. MIGRATION FROM PI 16.4 TO 16.5 ….………………………………………………...14](#Migration_From_PI)**

8. FAQ [16](#FAQ_)

[9. Notes 17](#_Toc482183497)

# INTRODUCTION

This document provides an overview of integration procedure for consumer care library in iOS mobile applications.

Consumer Care library provides set of standardized consumer care support features which can be used in all Philips apps.

# INTEGRATION

## Cocoa pod Integration

The easiest and preferred way to use these components is using Cocoapods (version 0.36.0 or newer). In your podfile, you have to add a source line for the Philips internal pod spec repository. So a minimal Podfile would look like this:

Source: ssh://tfsemea1.ta.philips.com:22/tfs/TPC\_Region24/CDP2/\_git/dcc-ios-consumercare-app

Or

<http://tfsemea1.ta.philips.com:22/tfs/TPC_Region24/CDP2/_git/dcc-ios-consumercare-app>

pod PhilipsConsumerCare, '~> 6.6.2

Please refer sample app to add dependent libraries.

Integrating application can get the framework files from the below path,

Source> ConsumerCareBinaries

## Library Integration

Git source path :

<http://tfsemea1.ta.philips.com:8080/tfs/TPC_Region24/CDP2/TEAM%20Sabers/_git/dcc-ios-consumercare-app>

1. Branch: Master
2. Check out the code from above path where in you can find sample app which depends on consumer care libraries.
3. Consumer care needs other libraries to build which are as below
4. pod ‘PhilipsUIKitDLS’
5. MBProgressHUD (MBProgressHUD.h, .m - <https://github.com/jdg/MBProgressHUD)> pod ‘MBProgressHUD’ , ‘~> 0.9.2’

e. pod ‘PhilipsPRXClient’

1. pod ‘PhilipsProductSelection’
2. AppInfra: pod ‘AppInfra’: Vertical app has to add AppInfra configuration via AppConfig.json, from **AppInfra** Component. Please refer Appinfra integration document.
3. pod 'UAPPFramework' : Vertical app has to add UAPPFramework for invoking all CoCo’s same follows for consumer care.

1. Please copy DigitalCareConfiguration.plist from library’s DigitalCareBundle.bundle to application’s folder and override the attributes.
2. Override the values in config file and define the features as per requirement.

## Prerequisites

1. Application need to know **CTN** number for each product used in app     Ex: HD8967/01 2.Along with sector and catalog. Example: **Sector:** B2C and **catalog**: CARE (Mandatory for Consumer care) Most of the apps will have above sector and catalog.

Above information is used for backend services and mostly uses PRX system.  Hence, please provide valid and complete CTN number as input otherwise consumer care features are not functional as expected.

# INITIALIZATION

1. App can invoke the library using below API’s

From PI16.4 all common components follow the microapp standards, so all the app has to use the uappframework standard API’s for invoking common components.

The invocation for ConsumerCare looks below e.g.:-,

**// setting hardcoded product list**

**PSHardcodedProductList \*hardcodedList =**

**[[PSHardcodedProductList alloc] init];**

**hardcodedList.catalog= CARE;**

**hardcodedList.sector= B2C;**

**[hardcodedList setHardcodedProductListArray:[NSArray arrayWithObject:[NSString stringWithFormat:@"%@",self.ctnField.text]]];**

**launchInput.productModelSelectionType = hardcodedList;**

**// Invoking CC with uApp API's**

**DCDependencies \*dcDependencies = [[DCDependencies alloc] init];**

**dcDependencies.appInfra = [AIAppInfraSingleton sharedInstance].sharedAppInfra;**

**self.dcInterface = [[DCInterface alloc]init];**

**self.dcInterface = [[DCInterface alloc]initWithDependencies:dcDependencies andSettings:nil];**

**DCLaunchInput \*launchInput = [[DCLaunchInput alloc] init];**

**launchInput.dCMenuDelegates = self;**

**UIViewController \*childViewController = [self.dcInterface instantiateViewController:launchInput withErrorHandler:^(NSError \*error) {**

**NSLog(@"error happened");**

**}];**

**[self.navigationController pushViewController:childViewController animated:YES];**

1. For supporting multiple product selection, we have introduced PhilipsProductSelection component. Vertical has to set the CTN’s in **PSHardcodedProductList** object as mentioned below.This is applicable for single as well as multiple products and depends on the array size consumer care will launch the multiproductselection library in consumercare .

Apart from this Vertical must set the SECTOR and CATALOG values as mentioned in the below enum’s.

Catalog Types Available:-

CATALOG\_DEFAULT,

CONSUMER,

NONCONSUMER,

CARE,

PROFESSIONAL,

LP\_OEM\_ATG,

LP\_PROF\_ATG,

HC,

HHSSHOP,

MOBILE,

EXTENDEDCONSENT

Sector Types Available:-

DEFAULT,

B2C,

B2B\_LI,

B2B\_HC

Ex:-

PSHardcodedProductList \*hardcodedList = [[PSHardcodedProductList alloc] initWithArray:[NSArray arrayWithObjects:@"HX9371/04",@"HF3330/01",@"HP6581/03",@"HP8656/03",@"S9031/26",@"SCF693/17",@"SCF145/06",@"FC6168/62",@"GC9550/02",@"PR3743/00", nil]];

hardcodedList.catalog=CARE;

hardcodedList.sector=B2C;

\* **DCDependencies** :-

All dependency to CC will passed via DCdependency , for CC AppInfra instance will be passed as dependency injection from integrating app,

**DCDependencies \*dcDependencies = [[DCDependencies alloc] init];**

**dcDependencies.appInfra = [AIAppInfraSingleton sharedInstance].sharedAppInfra;**

\* **DCLaunchInput** :-

Configurations required while launching CC has to be done here. CC requires the Menu delegate’s required to handle the menu buttons action handling to CC or Vertical app. The menu includes Support, Product, Social Button Menu’s.

**DCLaunchInput \*launchInput = [[DCLaunchInput alloc] init];**

**launchInput.dCMenuDelegates = self;**

Live Chat URL API can be configured using “**chatURL**” property of launchInput :

Eg:- If app country is china the china specific livechat url can be configured as below

**launchInput.chatURL = @"http://ph-china.livecom.cn/webapp/index.html?app\_openid=ph\_6idvd4fj&token=UniqueToken";**

**Important**:- Integrating App should append a unique value (token) to the URL as query parameter Livecom backend receives this value.

Menu configuration can be done based on the selection of country, Language, version etc.,

This Specific configuration can be handled via passing the plist path using below api with specific naming

Eg:- If app country or appversion or language etc based configurations app can pass their config file via api. This is helpful if app is in China and they don’t want facebook, twitter etc.

**launchInput.appSpecificConfigFilePath = [[NSBundle mainBundle] pathForResource:@"DigitalCareConfiguration\_china" ofType:@"plist"];**

\* **DCInterface** :-

Interface object provides the method for returning CC ViewController with parameter launchinput as below,

**UIViewController \*childViewController = [self.dcInterface instatiateView**

**Controller:launchInput withErrorHandler:^(NSError \*error) {**

**NSLog(@"error happened");**

**}];**

**[self.navigationController pushViewController:childViewController animated:YES];**

App version:-

Below API wil provide the current consumercare version

[DCConsumerCareVersion dcVersion];

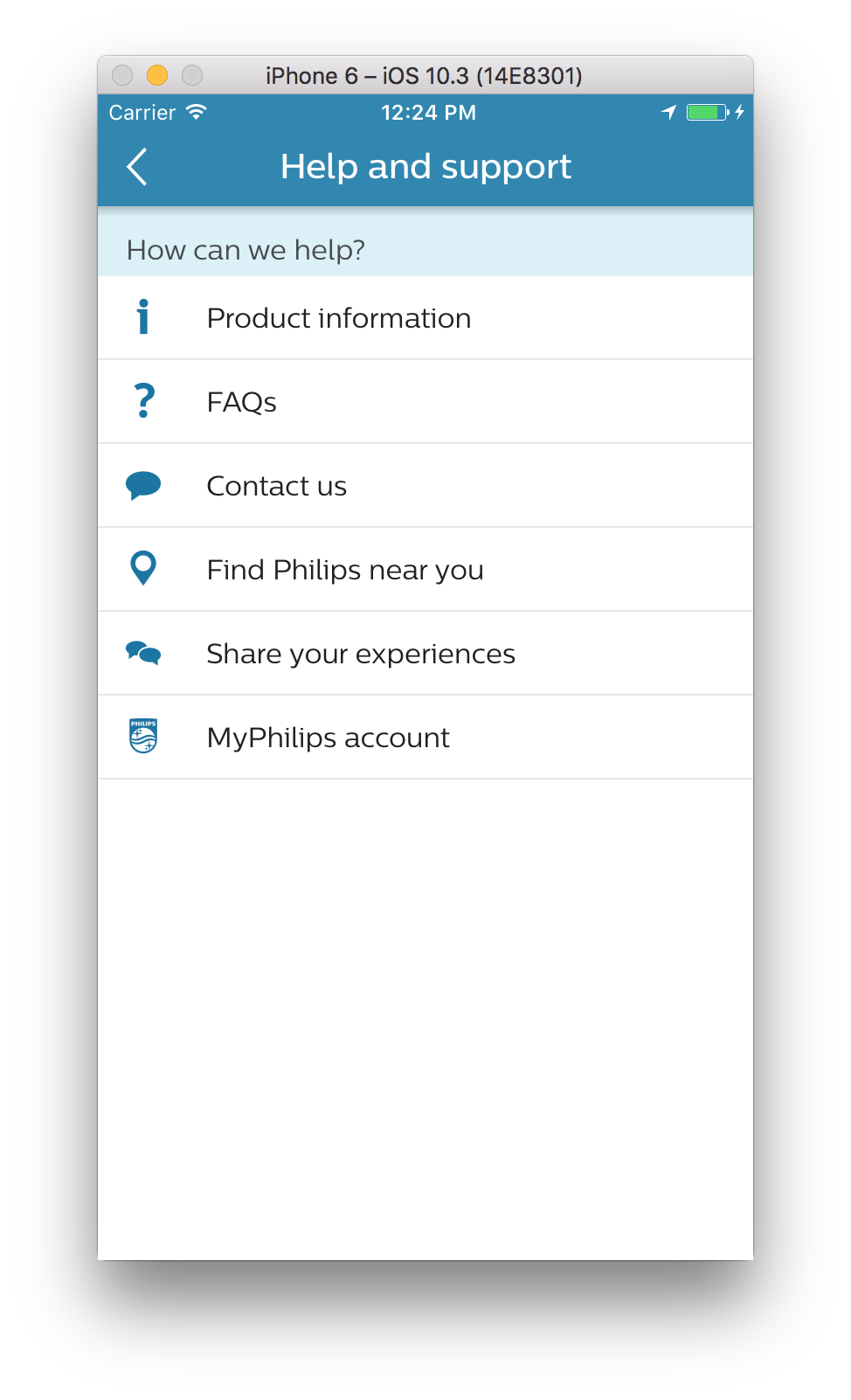
## Consumer Care Configuration

The application has to copy DigitalCareConfiguration.plist from library resource bundle and create one in app resource bundle by overriding the parameters present in it. The file contains the following,

1. UI resources like background image, background color and strings.
2. Facebook page id and twitter page name for respective Philips product page. If this is configured in service discovery the id’s will be taken from their service discovery, and if not available in SD it will refer from config file. Also in config file if there is no url the feature will be removed.
3. App store id for the app can be added in the FeedbackMenu for app feedback.
4. An array is defined for defining main menu title, main menu icons, product menu title and social provider list. Each menu contains default features supported by Consumer Care library. App can include only the required features by adding/deleting/modifying the array items.
5. Each title string in above arrays is expected to be a resource key and not the value.

**Important**: From PI17.5 onwards facebook, twitter , livechat configurations from config file will be deprecated and proprositions has to configure in Service deiscovery.

### Main menu configuration

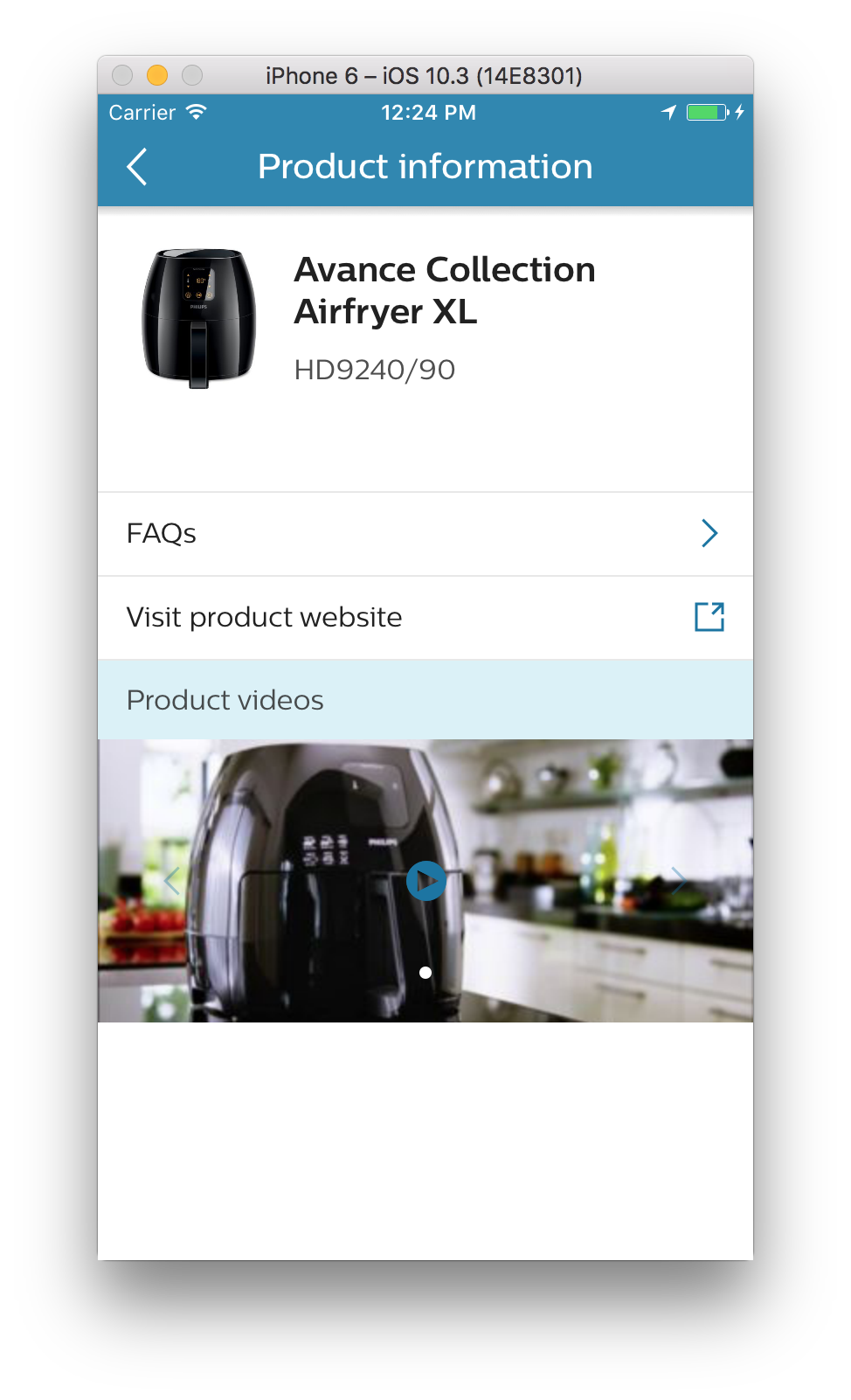


**ScreenBackgroundColor or BackgroundImage**

**IconName under Main Menu**

**Localisedkey under Main Menu**

### Product menu configuration

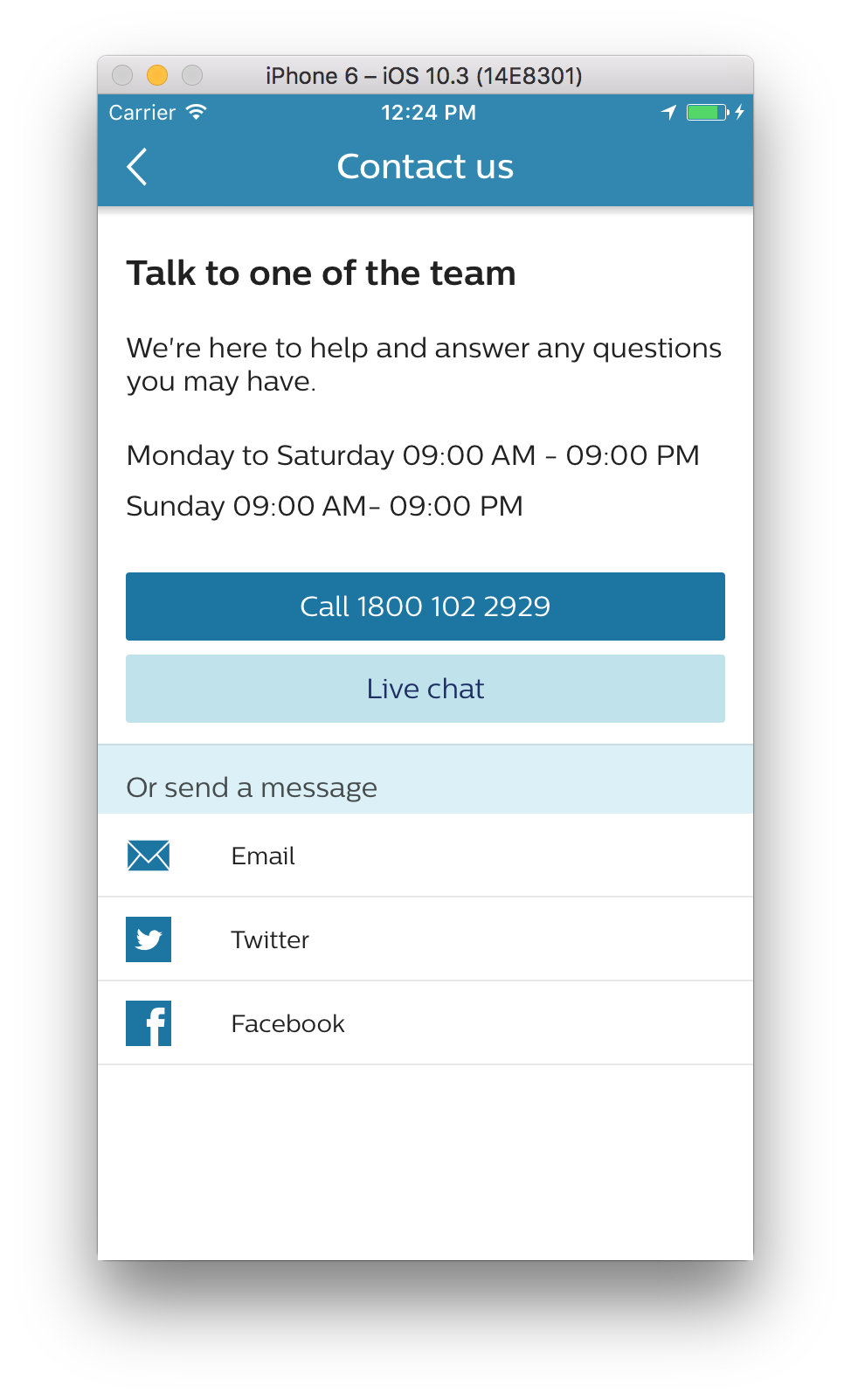


**LocalisedKey under**

**ProductMenu**

**Localisedkey under Product Menu**

### Social provider configuration



**LocalisedKey under**

**SocialListProviders**

# Vertical features customization

* 1. App can add its own button by passing valid string for button title and icon resource name in configuration file.
  2. Also application has to implement protocols like MainMenuDelegate,

ProductMenuDelegate, and SocialProviderDelegate which will be combined to **MenuDelegates** .

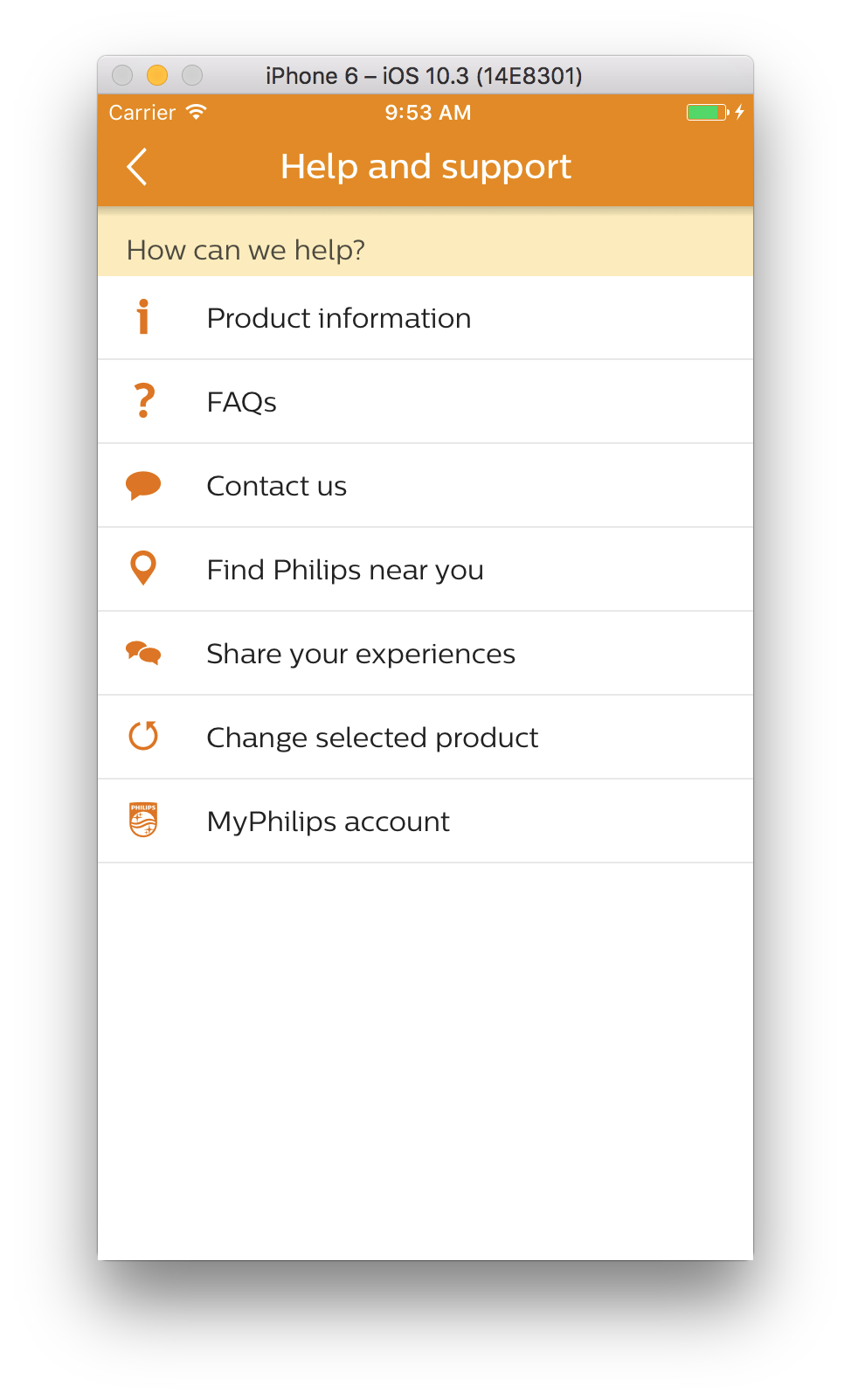
1. -(BOOL)mainMenuItemSelected:(NSString\*)item withIndex:(NSInteger)index; – Invoked on click of each button on support home page. This helps in configuring features on main/home page.
2. -(BOOL)productMenuItemSelected:(NSString\*)item; – Invoked on click of each button on product info page. This helps in configuring features on product info page.
3. -(BOOL)socialMenuItemSelected:(NSString\*)item; – Invoked on click of each social provider buttons in Contact us page. This helps in configuring list of social list providers. CC by default supports Facebook and Twitter.
   1. App is required to implement all these to take control on app side for the

menu items which is added by verticals. For Consumer Care menu item, library is supposed to take action and app should not ideally override this.

* 1. If app returns “true” then it means that app will take action on click of that

button and library will not perform anything in that case otherwise library will take action if app returns “false”.

Example: Air Fryer customization



# Navigation bar customization

PI16.4 release onwards ConsumerCare is returning the viewcontroller to integrating app’s. So the vertical app will be using their own NavigationControllers.

# Tagging

Tagging was completely handled by ConsumerCare with AppInfra tagging API’s, Please refer the AppInfra integration document for any clarification.

**Localization support**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| English | German | French (FR) | Simplified Chinese | Traditional Chinese | Portuguese (EU) |
| Russian | Arabic | Japanese | Dutch | Turkish | Danish |
| French-CA | Romanian | Spanish Mexican | Ukrainian | Portuguese (BR) | Vietnamese |
| Thai | Greek | Hebrew | Hindi | Finnish | Swedish |
| Italian | Polish | Spanish | Korean | Czech | English UK |
| Norwegian | Spanish (AR) | Bosnian | Bulgarian | Croatian | Estonian |
| Hungarian | Slovenian | Slovak | Latvian | Lithuanian |  |

# 

**7.** **Migration From PI17.3 to PI17.4**

1. Complete PUIKit support is removed and converted to DLS.

Configuring the theme:

Consumer Care is completely migrated to DLS so before invoking consumer care they have to invoke the DLS Theme. If application already using DLS theming do not need to call again.

[[UIDThemeManager sharedInstance] setDefaultThemeWithTheme:[UIDTheme new] applyNavigationBarStyling:YES];

1. Service discovery (SD) changes for propositions

a, **Facebook and Twitter** url’s are added in service discovery. Now verticals can add locale based facebook, twitter pages by configuring url’s in SD. Also providing backward compatibility, if no url’s in SD it will fallback to CC config file.

SD URL structure should look like below,

Facebook:- [https://www.facebook.com/**177538492375159**](https://www.facebook.com/177538492375159)

Twitter:- [https://twitter.com/**@PhilipsCare**](https://twitter.com/@PhilipsCare)

The above highlighted are the **pageid** and **pagename** which proposition has to fill based on locale.

**b, Live Chat:**

Live chat url also added in SD , and integrating app can have locale based chat url’s now. The first preference is setting chat url via API, if not it will fall back to SD and at last to config file.

Each business can get the livechat url's from CDLS server where proposition has to give the **locale** and **subcategory** as highlighted in below example ,

[https://www.philips.com/prx/cdls/B2C/**en\_GB**/CARE/**CC\_SMART\_MONITORS\_SU**.querytype.(fallback)](https://www.philips.com/prx/cdls/B2C/en_GB/CARE/CC_SMART_MONITORS_SU.querytype.(fallback))

Few of the live chat url’s from some locale’s are not secure , so App Transport Security will block those domains and not allow to load. We already raised a ticket to server team for the fix. CC will show error alert in chat pages with support not available. Either Proposition has to whitelist those domains in their app info.plist to work till it fix.

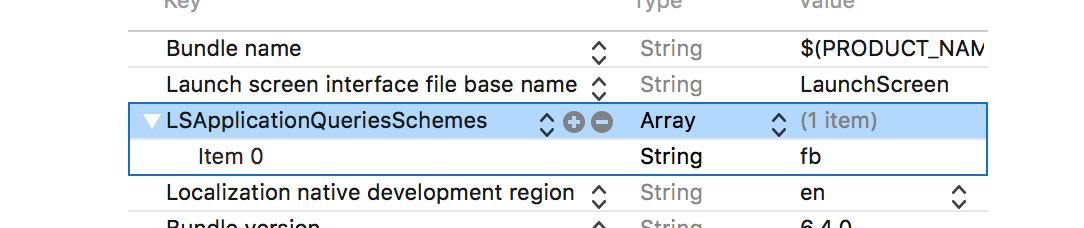
Below are the few of known chat domains which are not secure which proposition has to add in the AppTransportSecurity in info.plist,

[https://apps.caas.com](https://apps.caas.com/Philips/chat/?provider=philips&language=en&country=GB&group=PERSONAL_CARE&category=TOOTHBRUSH_HEADS_CA&sub-category=DIAMONDCLEAN_SU2&ctn=PARAM_CTN), [https://ph-eur.livecom.net](https://ph-eur.livecom.net/5g/ch/?___________________________________________________________=&aid=ozbYgjYIAAA%3D&gid=7&skill=undefined&tag=PERSONAL_CARE_GR&cat=&chan=LWC;LVC;LVI&fields=&customattr=Group%3APERSONAL_CARE_GR%3B%20Category%3A%3B%20Sub-category%3A%3B%20CTN%3A%3B%20Country%3ADK%3B%20Language%3Ada&sID=iQ5kwb8nAwM%3D&cID=6gXaY0finwE%3D&lcId=CONTACT_DA_DK&url=http%3A%2F%2Fjson.parser.online.fr%2F&ref=) etc…

Note:- If integrating app want only SD based configuration they can remove facebook page id ,twitter pagename and live chat from CC config file which will remove the buttons if data not available in SD.

1. Facebook Support

Facebook will support only via pageid. Integrating app has to add the query scheme in the app info.plist file for opening facebook if facebook app is installed in device. Please find the below query scheme to be added in plist as in the below screenshot,



# FAQ

1. **How about customizing fonts and sizes?**

We use Central Sans Font as per Philips design guidelines. Both fonts and sizes cannot be configured as of now.

1. **Can we add our own buttons inside consumer care?**

Yes. However we have restricted this to main menu screen, product menu and list of social service providers list.

1. **Does Consumer care support App Tagging?**

Yes. ConsumerCare uses the tagging provided by AppInfra.

1. **Where do I find my product information?**

Each business has to provide complete ctn number including slash

Ex: HD9240/90

With this please check below link which helps in finding out product tree information like catalog, sector and ctn. Please add relevant ctn and locale number as below PRX Product Summary request

<https://www.philips.com/prx/product/B2C/en_GB/CARE/products/HR2106/91.summary>

Please replace locale to check whether product data is available in corresponding countries.

1. **Does Consumer care support multiple products?**

Yes, it supports multiple products from 16.1 release which has version 4.0.0 onwards.

Currently app can send hardcoded list of products as input after which consumer care will take care of selection of product.

1. **Does Consumer care support Facebook and Twitter from Service discovery?**

Yes, it supports from Service Discovery as the first preference. If data not available it will fallback to local config file for this PI. If local config file data is not their the respective buttons will be removed. From PI17.5 onwards local config file support will be deprecated.

1. **Does Consumer care support LiveChat, from Service discovery?**

Yes, it supports from Service Discovery , but the first preference is via API, then the second preference goes to SD. If data not available it will fall-back to local config file for this PI. If local config file data is not their, the respective buttons will be removed. From PI17.5 onwards API and local config file support will be deprecated.

# 9. Notes

1. Registration is developed as separate library project. App is expected to invoke registration library on click of registration button inside consumer care support page.
2. Please refer sample application for more details.
3. Consumer Care 6.6.0 version will give support from iOS 9 and greater.
4. Remove the App tagging API’s which was their in <5.0.0 versions of consumercare, as it is taken cared by AppInfra API’s.
5. It is very important that ,from iOS 10 onward’s integrating app has to add privacy messages in info.plist. Consumer care uses Camera and Photolibrary for posting images to twitter. So vertical can add the key and messages like below,

1, *Privacy - Photo Library Usage Description        -           App used to access your photo library to post it in Twitter Page*

*2, Privacy - Camera Usage Description                  -           App used to take photos of yours to post it in Twitter Page*

Note:- Above strings can be added by vertical as per the use case and sample we mentioned. If not added it can lead to app rejection from Apple from iOS 10 onwards.

Consumer care component consumes PRX data for most of the features. From **iOS** **9** onward’s APPTransportSecuritySettings is blocking **NONHTTP** urls. Few LiveChat url’s are not secure from PRX – CDLS Server, we already raised request for converting all the chat url’s to **secure**. Till they fix it, all the integrating app’s has to whitelist the http domain’s or AllowArbitrayLoads in their app info.plist for the respective url’s as mentioned in Migration from PI16.3 to 16.4 header.

1. China release related docs attached below.
2. For china livechat url we exposed the API. Integrating App should append a unique value (token) to the URL as query parameter Livecom backend receives this value. If the token is matched in the database *(in other words, if the user has already used the live chat inside the app),* Livecom will identify this as an existing "contact" and make his history accessible to the agent. On the APP side, the user will be able to go back through his previous history.

