Open Issues and/or Actions

“No open issues” in this section.

Table of Contents

[1 Document Introduction 3](#_Toc475381538)

[1.1 Purpose 3](#_Toc475381539)

[1.2 Scope 3](#_Toc475381540)

[1.3 References 3](#_Toc475381541)

[1.4 Terminology & Abbreviations 3](#_Toc475381542)

[2 Overview 4](#_Toc475381543)

[3 General Requirements 4](#_Toc475381544)

[3.1 Compliance Requirements 4](#_Toc475381545)

[3.1.1 Compliance Domain 5](#_Toc475381546)

[4 Function/Feature Requirements 5](#_Toc475381547)

[4.1 Creating the Reference App 5](#_Toc475381548)

[4.2 AFW separation from BaseApp source and packaging into a CoCo (managed library) incl. MVP flowmanager improvements 11](#_Toc475381549)

[4.3 Convert BaseApp into ReferenceApp 12](#_Toc475381550)

[4.4 Healthy Sleep Solution (PowerSleep & BrightEyes) support for platform use 13](#_Toc475381551)

[4.5 Aligned ALM solution across VS Platform (TFS technology, I2M compliant) 13](#_Toc475381552)

[5 Interface Requirements 13](#_Toc475381553)

[5.1 User Interfaces 14](#_Toc475381554)

[5.2 Technical Interfaces 15](#_Toc475381555)

[6 Revision History 15](#_Toc475381556)

[7 Approval 16](#_Toc475381557)

# Document Introduction

## Purpose

This Software Requirements Specification (SwRS) describes the software system requirements, the functional behavior of the application and its software items / units as well as nonfunctional requirements, design constraints, and other factors necessary for design and development.

## Scope

The document describes requirement for mobile application framework and base application (built on app framework). The document does not contain requirements for common components which are integrated in mobile app framework.

## References

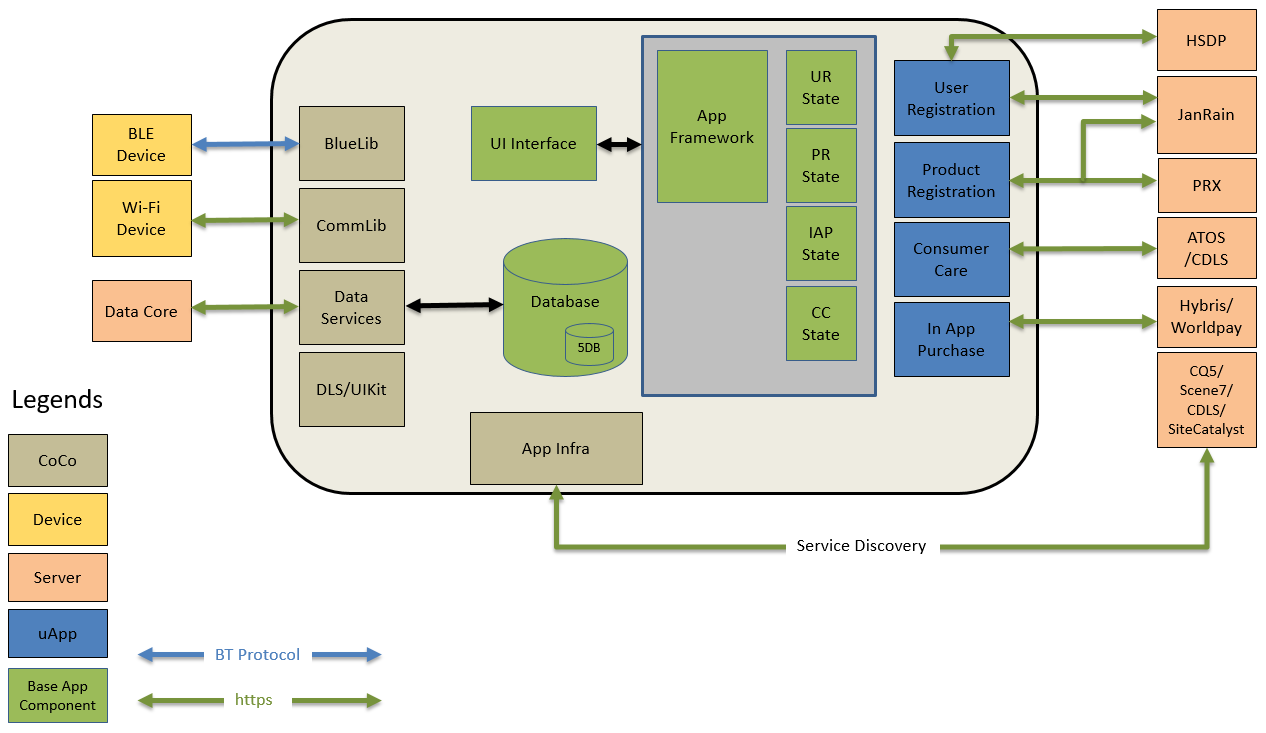
Not applicable.

## Terminology & Abbreviations

| **Terminology & Abbreviations** | **Description/Definition** |
| --- | --- |
| App | Application |
| UI | User Interface |
| AppFWK / AFW | Application Framework |
| CoCo | Common Component |
| AF | Amalgam Feature |
| MVP | Minimum Viable Product |

# Overview

Reference app is a base platform used by all propositions to develop and maintain features both generic and specific to the propositions. App infra, Data Gateway, App FWK, and UIToolkit allow Apps to be created from uApps and UI-less services/libraries.



# General Requirements

## Compliance Requirements

| **Compliance Domain** | **In scope** | **Rationale** |
| --- | --- | --- |
| Internal Control |  |  |
| Information Security |  |  |
| Legal |  |  |
| Payment Card Industry |  |  |
| Product and Service Security |  |  |
| Quality and Regulatory | Y | As per the IEC16304 guidelines. |
| Sustainability |  |  |
| Tax |  |  |

### Compliance Domain

| **ID** | **Requirement Description** | **Category** |
| --- | --- | --- |
| 1 | ARCHITECTURE | IEC 62304, 3.3: organizational structure of a SYSTEM or component. |
| 2 | SOFTWARE ITEM | IEC 62304, 3.25: any identifiable part of a computer program. |
| 3 | SOFTWARE SYSTEM | IEC 62304, 3.27: integrated collection of SOFTWARE ITEMS organized to accomplish a specific function or set of functions |
| 4 | SOFTWARE UNIT | EC 62304, 3.28: SOFTWARE ITEM that is not sub-divided into other items. |
| 5 | SOUP | IEC 62304, 3.29: Software of Unknown Provenance.  SOFTWARE ITEM that is already developed and generally available and that has not been developed for the purpose of being incorporated into the MEDICAL DEVICE (also known as “off-the-shelf software”) or software previously developed for which adequate records of the development PROCESSES are not available. |
| 6 | SYSTEM | IEC 62304, 3.30: integrated composite of one or more of the PROCESSES, hardware, software, facilities, and people, that provides a capability to satisfy a stated need or objective |

# Function/Feature Requirements

## Creating the Reference App

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-1 | Base App | | |
| As part of this feature :  Splash screen is created in basic app,  Welcome/Introduction screen is created in basic app,  Home screen is created in basic app,  Settings screen is created in basic app,  Help screen is created in basic app,  User can navigate to Splash >> Welcome (if user is not registered) >> Home,  User can navigate from Home to Setting or Help and vice a verse,  Contents on Splash, Welcome/Introduction, Home, Settings and Help can be customized by vertical apps. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-4 | uApp Configuration | | |
| As part of this feature :  Creating a developer interface which can be used by all CoCoS in their respective common components to integrate with to app framework based on the architecture guidelines  Using uApp configuration component from App Infra to configure the CoCoS in app framework. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-5 | UI Plugin/Pluggable Architecture | | |
| As part of this feature :  Configuration,  Launching mechanism,  Initialization,  Action bar handling,  Back key handling,  Dependency injection are handled,  Android and iOS versions released the micro app framework as library and named it as uAppFramework.  TICS overall quality score was improved to B & above. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-6 | Architecture Runway: Dependency Framework for Android | | |
| As part of this feature :  Understanding the Dagger 2 framework and setting it up within the reference app framework context was achieved. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-7 | Data Sync MVP | | |
| As part of this feature :  An interface for a Data sync CoCo that will be used to send the application data towards the Data core back-end. This included  Generalizing the data structure that needs to be sent to back-end,  Storing data locally,  Sending data to back-end,  Syncing data between back-end and local storage,  Notifying UI, when local data changes and  Taking care of the network unavailability for syncing. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-8 | Unit Test coverage improvements for PI 16.4 | | |
| As part of this feature :  48% code coverage has been achieved till now. Target is to achieve 80%. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-9 | Base App Improvements | | |
| As part of this feature :  Integrating with connectivity components (DICOMM and BLUELIB) in the reference app was achieved for establishing the connection with physical devices.  After completing the integration, build was successfully made and the reference app was launched without any crashes/regressions. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-11 | Platform Integration Support for verticals | | |
| As part of this feature :  Effort spent towards supporting verticals was captured, so that reference app team learn on the capacity needed to be reserved for such activities in future. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-12 | Tech Backlog: iOS 10 & Swift 3 support | | |
| As part of this feature :  Reference app was enabled to run on iOS 10 OS,  Migrating the syntax of the swift language code from Swift 2 to Swift3 was achieved. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-13 | Code Refactoring for review comments | | |
| As part of this feature :  Review comments from architects for  Back handling, code styling and call back listener, home activity refactoring, all the CoCoS imports in one class are captured and implemented, so that the reference app can deliver a robust code base in both Android and iOS versions. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-14 | Compile Time configuration of CoCo's | | |
| As part of this feature :  An option for enabling the propositions to build only the required CoCo libraries and to disable all the other unwanted libraries from getting built was provided.  In iOS, this is achieved by the preprocessor directives that will enable/disable the code from getting compiled,  In Android, using annotations unused CoCo interfaces will be stubbed. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-15 | Architectural Runway: UI Configuration for Tabbed Menu | | |
| As part of this feature :  Providing the option in reference app for propositions to configure either the tabbed interface or hamburger menu interface based on their requirements. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-16 | Integration of MicroApps | | |
| This activity includes the integration of the latest releases of all CoCo's are integrated into reference app for every sprint.  On Android, this is mostly automated. However, supervision is still required.  On iOS, this is a manual activity. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-17 | As system integration tester I would like to ensure all the COCO's are integrated with the base app | | |
| As part of this activity, complete manual integration testing of reference app along with all the integrated CoCoS are performed. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-18 | KT to new developers | | |
| As part of this activity, knowledge transfer on app framework and reference app for new developers were provided. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-19 | Base App UI modifications | | |
| As part of this feature :  Capability of configuring the backend environments dynamically is provided. This helps in testing the reference app in Dev or staging environments. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

## AFW separation from BaseApp source and packaging into a CoCo (managed library) incl. MVP flowmanager improvements

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-22 | Organize the code base (App framework separation from base app code) in a manner which is easily extendable by the users so that I can onboard more proposition to consume the base app/Reference app code | | |
| As part of this feature :  Multiple repos were streamlined i.e. App Framework\_Android (containing BaseApp and Flow manager) and uAppFramework\_Android (containing uAppFramework), plus equivalents for iOS, were changed to   * rap-android repo and rap-ios repo, including the ReferenceApp 2.0 source code (merger of bap and rap) * afw-android repo and afw-ios repo, including: Flow Manager, uAppFramework, as a 'new' CoCo   Notified all the relevant teams (CoCo + ReferenceApp) to update their microApps to incorporate the above listed changes for updating their dependencies and automated tests accordingly. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-23 | Flow Manager Improvements (Threading logic, Error Handling, Unit-Test Coverage) | | |
| As part of this feature :   * Threading logic was improved by parsing the AppFlow JSON and moving it to the background thread - off the main thread. This has helped in avoiding the long-running processes on the main thread in android, which results in application crash. * Error Handling has been improved by specifying a formal specification of the AppFlow JSON definition (POJO etc.), and implemented a validation step within the application of the application's AppFlow JSON instance against the said AppFlow JSON definition/schema. Also, run-time exceptions were shown during execution, when the flow manager runs into invalid or missing states. * Unit-test coverage has been improved, however target to reach 80% is yet to be achieved. * Back-button handling has been improved extending the flow manager to support configurable back-button handling, via a new "Back" event, and a stack-of-states in the Flow Managers (keep track of Previous, Current, and Next states), and configurable mapping of a Back event to the desired state-transition. * APIs has been improved by making suitable API extensions. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-24 | Improve Code coverage and TICS score | | |
| As part of this feature :  Overall code quality of reference app for both iOS and Android was improved, so that the quality matrix report met the CDP2 coding standards. This is an ongoing activity. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

## Convert BaseApp into ReferenceApp

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-25 | As a product owner I want to upgrade BASE APP to REFERENCE APP so that we have a fully automation tested mobile App which acts as a reference for new propositions(Android Only) | | |
| As part of this feature :   * Merging repositories and all applicable branches, * Making sure that the latest and greatest BaseApp source code is merged into the existing ReferenceApp (rap) repository, * Converting BaseApp flow to ReferenceFlow using the Flowmanager configuration from the new App framework, * Integrating the MobileFE, Connectivity and BackEnd components in the new reference app, which includes building screens to test the E2E Use Cases. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-26 | BASE App to integrate with the latest builds from CoCo dcc, dsc | | |
| As part of this feature :  All the latest changes made in the CoCoS were successfully integrated in reference app. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-27 | Update Reference APP to integrate with the latest CoCo builds | | |
| As part of this feature :   * All CoCoS were assessed for any breaking changes: iap, usr, prg, dcc, ail, uit, dsc, bll, cml, * Optionally, monitoring the CoCo VERSION file and CHANGELOG file, or equivalent file for identifying the actual breaking changes, * Making required changes in the reference app (Android and iOS ReferenceApp) for overcoming the breaking changes for keeping the reference app build and tests green. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

## Healthy Sleep Solution (PowerSleep & BrightEyes) support for platform use

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-28 | Support propositions (Power sleep) during the 17.1 PI (effort capped) | | |
| As part of this feature :   * 30% of team capacity was spent in 17.1 PI for supporting propositions (notably powersleep, some vitaskin), * Having no open defects as part of the support activity that was blocking the week 3 and week 9 releases of Powersleep, * Also, implemented the OneRoof activities along with the support provision to propositions for resolving the issues. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

## Aligned ALM solution across VS Platform (TFS technology, I2M compliant)

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-29 | Code archive migration to TFS-GIT | | |
| As part of this feature :  All GIT code archives owned by team Amalgam were migrated to TFS-GIT, with help of team Expendables. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

# Interface Requirements

The integration interface should be as per the integration guidelines published in <https://confluence.atlas.philips.com/display/BA/BaseApp+Integration+Guidelines> link.

## User Interfaces

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-2 | Base App : Product Registration flow integration in Base App | | |
| As part of this feature,  Navigation flow/point to product registration from reference app is provided.  The navigation flow to product registration in reference app should be configurable i.e. if verticals don't need the product registration, then they should be able to take the flow out of reference app easily. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-3 | Base App: In App Purchase flow integration in Base App | | |
| As part of this feature,  Navigation flow/point to In App Purchase from reference app is provided,  Navigation flow/point to buy a product using In App Purchase is provided,  Navigation flow/point to view purchase history from In App Purchase is provided,  Navigation flow to In App Purchase is made configurable. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-10 | Tech Backlog: Android N fixes | | |
| As part of this feature :  Reference app was made compatible with Android N release by ensuring no changes in UI behavior when compared to the behavior when it is in full screen and by making the UI to be re-sizable on split window, so that the propositions can consume the same.  This is done as Android 'N' version supports multi window feature where in two separate apps can be viewed and operated on the mobile screen.  Also, checking the reference app for crashing on making changes to the user permissions at runtime was made. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-20 | Support Retailer purchase flow -Basic | | |
| As part of this feature :  For enabling the reference app for using the service discovery for identifying the INAPP purchase mode and in turn provide the options to customers to buy Philips products.  Since, Hybris is not enabled and when the customer clicks on "Philips Shop button" on Base APP (The tab name may vary based on Proposition decisions) the app fails.  Hence, displaying only the product list as shown in internal Philips shop is currently shown to the customers and the option of adding the selected products to cart and buying it will be provided when Hybris is enabled. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |

| **ID** | **Function** | | |
| --- | --- | --- | --- |
| AF-21 | Localization of Reference APP application strings. | | |
| As part of this feature :  Localization was enabled in reference app for supporting different languages. As a result of this, when the user installs the reference app then based on the selected country and language in the mobile settings, the screens content should tailor accordingly. This means that the language and content shown on the reference app screens should be based on the selection made in mobile settings.  Documentation of this was also delivered, which the propositions can access for understanding the working of localization in reference app.  **NOTE:** Individual CoCoS may ignore the language setting and only rely on the selected country to display the content. Under these circumstances the behavior of the screens will not be in line with what reference app, because the language setting would have been ignored and the countries default language would have been selected, For example FAQ's in consumer care component may support for the country only in the local language and not in english. These scenarios will not be captured from the reference app team. | | |
| **ID** | **Quality Aspect Requirements** | **Quantification** | **Category** |
|  |  |  |  |

## Technical Interfaces

The technical interface should be as per the guidelines published in <https://confluence.atlas.philips.com/display/BA/BaseApp+Architecture+Overview> link.

# Revision History

| **Version** | **Date** | **Author** | **Description of Change** | **Reason for Change** |
| --- | --- | --- | --- | --- |
| 0.1 | 2017-FEB-20 | Ramaprasada Udupa. S | Draft version | Draft version |
|  |  |  |  |  |
|  |  |  |  |  |

# Approval

| **Name** | **Role / Function** | **Date** (YYYY-MON-DD) | **Signature** |
| --- | --- | --- | --- |
| Deepthi Shivakumar | Technical Architect | 2017-FEB-20 |  |
|  |  |  |  |