|  |  |
| --- | --- |
|  |  |

App SDK – System Architecture Document.

# Abstract

This document describes in detail the Application SDK home area system architecture including the template, application, and service management.

Contents

[1 Introduction 2](#_Toc497222527)

[1.1 Revision History 2](#_Toc497222528)

[1.2 Purpose and Scope 2](#_Toc497222529)

[1.3 Target Audience 2](#_Toc497222530)

[1.4 Abbreviations and Acronyms 2](#_Toc497222531)

[1.5 Terms 2](#_Toc497222532)

[1.6 References 3](#_Toc497222533)

[2 System Architecture 4](#_Toc497222534)

[2.1 Overview 4](#_Toc497222535)

[2.2 System Architecture Diagram 4](#_Toc497222536)

[2.3 Component Specifications 5](#_Toc497222537)

[2.3.1 Application Manager 5](#_Toc497222538)

[2.3.2 Template Manager 5](#_Toc497222539)

[2.3.3 Service Registry 5](#_Toc497222540)

[2.3.4 Metadata store 6](#_Toc497222541)

[2.3.5 CLI 6](#_Toc497222542)

[2.3.6 Portal UI 6](#_Toc497222543)

[3 Non-Functional Requirements 7](#_Toc497222544)

[3.1 Overview 7](#_Toc497222545)

[3.2 Security and control 7](#_Toc497222546)

[3.3 Performance and availability 7](#_Toc497222547)

[3.4 Back-up and contingency 7](#_Toc497222548)

[4 Appendix A – PBA Reference Model 8](#_Toc497222549)

[4.1 PBA Model 8](#_Toc497222550)

[5 Appendix B – Open Issues 8](#_Toc497222551)

# Introduction

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Name** | **Information** |
| 0.1a | 2017-10-02 | Mohamed Ibrahim | Initial draft. |

## Purpose and Scope

The purpose of this document is to present a comprehensive reference for the App SDK Home area’s functional and technical system architecture.

## Target Audience

This document is primarily intended for the following audience:

* AIA Architecture, Design, and Management Teams.
* AIA Development and QA Teams.
* External Stakeholders interested in AIA Platform.

## Abbreviations and Acronyms

|  |  |
| --- | --- |
| **Term** | **Definition** |
| PBA | Platform Based Appliance. |
| ECN | Ericsson Corporate Network. |
| AIA | Analytics Implementation Architecture. |

## Terms

|  |  |
| --- | --- |
| **Term** | **Definition** |
| AIA Analytics Application | AIA Analytics applications are typically created by any user who has the ‘Application Flow Developer’ role and they are based on a published AIA Template.  An application basically consists of a structured directory of source code based on an existing template in addition to a ‘Flow.xml’ file which contains the user’s business logic (processing steps definition).  Applications consist as well from a PBA descriptor file referencing a runnable docker image for certain technologies/services. |
| Template | AIA realizes a model driven approach in building analytics applications by using Templates.  A Template is the foundation of AIA batch and stream processing pipelines and based on which applications are created.  Templates define a standard structure consisting of source code, data sinks, data sources and processing technologies. |
| Service | AIA platform adopts a micro-services architecture by supporting a set of interdependent ‘services’ inside the service registry component.  Services can be used during template creation to define the different connector’s data sinks, sources and processing technologies and related attributes. They also represent the various cloud services which can be deployed on the platform such as messaging as a service, Kafka as a service…etc  A service PBA definition file references a runnable docker image. |

## References

|  |  |
| --- | --- |
| **Doc.** | **Location / URL** |
| Analytics Implementation Architecture Proposal | <https://confluence-nam.lmera.ericsson.se/download/attachments/131989770/Analytics%20Implementation%20Architecture.docx?version=1&modificationDate=1462459517000&api=v2> |
| SDK Architecture Portal | <https://confluence-nam.lmera.ericsson.se/display/DA/SDK+Architecture> |
| Metadata Service Confluence Page | <https://confluence-nam.lmera.ericsson.se/display/DA/Metadata+Service> |

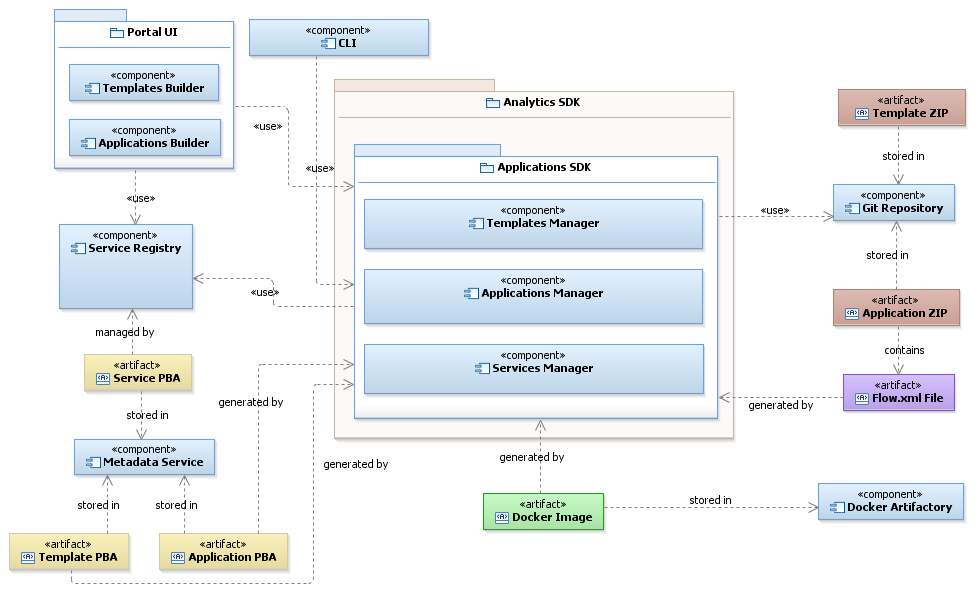
# System Architecture

## Overview

The purpose of this chapter is to present a comprehensive explanation of the Application SDK system architecture as well as the relationship between the various functional blocks.

The below system architecture diagram highlights the general interactions between the different AIA platform components / functional blocks and the Application SDK.

## System Architecture Diagram



*Figure 1 – Application SDK System Architecture Diagram.*

The above diagram describes the main functional blocks and artifacts which the AIA Application SDK consists from as well as the relationship between them.

In the next section, there will be a description for each component/block role.

## Component Specifications

### Application Manager

This component is responsible for functionalities related to Applications Management such as:

* Application Creation.
* Applications Retrieval.
* Application Extend.
* Application Publish.
* Applications Delete, Cleanup and unpublishing.

It’s also responsible of creating the application’s ‘Flow.xml’ (BPS 2.0) file and later storing the application’s ZIP archive in Git/Nexus and computing the appropriate version.

The Application Manager can be used via invoking the related interface REST End-point or via an embedded library mode.

### Template Manager

The Template Manager component is responsible of all the functionalities related to Templates Management in AIA platform such as:

* Template Creation.
* Templates Retrieval.
* Template Extend.
* Template Publish.
* Templates Delete, Cleanup and unpublishing.

The Template Manager can be used via invoking the related interface REST End-point – as described in Chapter 5 – or via an embedded library mode.

### Service Registry

The Template Manager component is responsible of managing all the supported technologies definition in AIA platform. Typically, the core AIA team will be responsible of adding a new service definition file (PBA) corresponding to each published technology docker container/image.

It’s also responsible of validating the integrity of the services defined within the registry which means to check whether a valid docker container exists or not and which is linked to a certain service definition and so on.

The service registry component offers the following functions:

* Services Definition and creation.
* Services Retrieval.
* Services available versions retrieval.
* Deleting services.

### Metadata store

The Metadata service component is basically a customized interface for an underlying database – MongoDB is the current implementation – to store and manage any metadata information in AIA platform.

Currently, it’s mainly used to store the various PBA data around Applications, Templates, and Services. It offers various functionalities on top of the regular storage such as:

* Custom PBA input validation based on the latest PBA reference schema definition.
* Searching by PBA JSON property attributes’ paths.
* Update specific PBA JSON attribute by property path.
* Schema existence validation.
* Encapsulates the dependency resolution information between applications, services (and each other’s), and templates.

### CLI

A command line tool offering a comprehensive set of commands/utilities like the functionalities offered in the GUI.

### Portal UI

This is a web based GUI application for AIA. It is composed of the following components:

* Template Builder.
* Application Builder.

Please refer to Appendix A for more information about the PBA schema model and sample PBA JSON files for templates, applications, and services.

# Non-Functional Requirements

## Overview

This chapter discusses the various non-functional requirements aspects which the AIA platform should provide.

## Security and control

* All REST calls between any client and the AIA back-end services should be via HTTP/S.
* SSL based certificates should be used to authenticate the communication with any external entity.
* LDAP user authentication & SAML 2.0 tokens (including roles) shall be used.

## Performance and availability

* The AIA platform should achieve an availability level (including the main components such as Portal and back-end REST services) of 99.9 %.
* The platform will not be available during Maintenance and Upgrade time windows.
* Applications and Templates publishing operation should not exceed 2 minutes from beginning to end.
* Services deployment operation should not exceed 5 minutes from beginning to end.

## Back-up and contingency

* The PBA data inside the Metadata service should be backed up on regular basis (configurable frequency) to a DR server.
* Same above rule should be applied to the published applications and templates data archives stored in the Git/Nexus repository.
* Users local workspace data is out of scope.

# Appendix A – PBA Reference Model

## PBA Model

Gerrit Repository Path for master PBA model definition:

**AIA/com.ericsson.component.aia.sdk/sdk-model**

# Appendix B – Open Issues

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Issue Description** | **Status** | **Related JIRA Ticket #** |
| 1 | Some attributes such as ‘drop-malformed’, ‘skip-comments’ and ‘quote’ are not provided neither from the Template definition nor the user is able to manually provide them via the UI. | Open |  |
| 2 | Application and Template Versioning Validation. | In Progress |  |
| 3 | Verify the contents of the AIA Application and Template ZIP archives. | Open |  |
| 4 |  |  |  |