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

App SDK – Use Cases Document.

# Abstract

This document describes in detail the Application SDK home area requirements and use cases including the template, application, and service management functionalities.

Contents

[1 Introduction 2](#_Toc497222443)

[1.1 Revision History 2](#_Toc497222444)

[1.2 Purpose and Scope 2](#_Toc497222445)

[1.3 Target Audience 2](#_Toc497222446)

[1.4 Definitions and Acronyms 2](#_Toc497222447)

[1.5 References 2](#_Toc497222448)

[2 Use Cases 3](#_Toc497222449)

[2.1 Overview 3](#_Toc497222450)

[2.2 Use Cases Diagram 4](#_Toc497222451)

[2.3 SDK Developer Role Use Cases 5](#_Toc497222452)

[2.3.1 Create Template - UC\_APP\_SDK\_SD\_001 5](#_Toc497222453)

[2.3.2 Extend Template - UC\_APP\_SDK\_SD\_002 6](#_Toc497222454)

[2.3.3 Register Template - UC\_APP\_SDK\_SD\_003 7](#_Toc497222455)

[2.3.4 Unregister Template - UC\_APP\_SDK\_SD\_004 8](#_Toc497222456)

[2.3.5 List Templates (View Public Catalogue) - UC\_APP\_SDK\_SD\_005 9](#_Toc497222457)

[2.3.6 Delete Template - UC\_APP\_SDK\_SD\_007 9](#_Toc497222458)

[2.3.7 Register Service - UC\_APP\_SDK\_SD\_001 10](#_Toc497222459)

[2.4 Application Flow Developer Role Use Cases 11](#_Toc497222460)

[2.4.1 Create Application - UC\_APP\_SDK\_AFD\_001 11](#_Toc497222461)

[2.4.2 Extend Application - UC\_APP\_SDK\_AFD\_002 12](#_Toc497222462)

[2.4.3 Unpublish Application - UC\_APP\_SDK\_AFD\_003 13](#_Toc497222463)

[2.4.4 Publish Application - UC\_APP\_SDK\_AFD\_005 14](#_Toc497222464)

[2.4.5 List Applications (View/Search Public Catalogue) - UC\_APP\_SDK\_AFD\_006 15](#_Toc497222465)

[2.5 Solution Engineer Role Use Cases 16](#_Toc497222466)

[2.5.1 Compose Application - UC\_APP\_SDK\_SE\_001 16](#_Toc497222467)

[3 Appendix A – Open Issues 17](#_Toc497222468)

# 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 centralized comprehensive reference for the App SDK Home area’s business and technical requirements.

## Target Audience

This document is primarily intended for the following audience:

* Core AIA Architecture/Design/Management Team.
* Core AIA Development and QA Team.
* External Stakeholders interested in AIA Platform.

## Definitions and Acronyms

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

## References

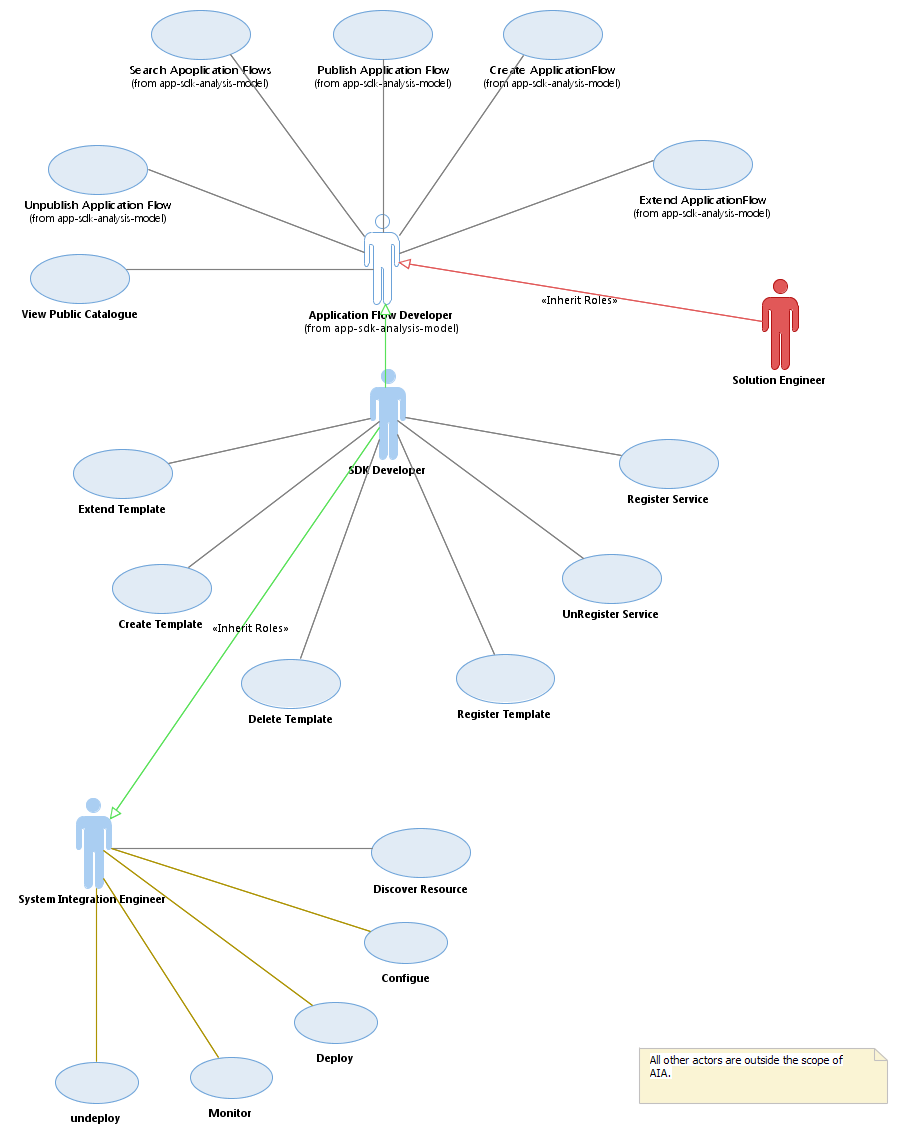
|  |  |
| --- | --- |
| **Doc.** | **Location / URL** |
| Analytics Implementation Architecture Proposal | <https://confluence-nam.lmera.ericsson.se/download/attachments/131989770/Analytics%20Implementation%20Architecture.docx?version=1&modificationDate=1462459517000&api=v2> |
| SDK Architecture Portal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

# Use Cases

## Overview

The purpose of this chapter is to present in detail the various use cases supported by the Application SDK home area which are mainly related to: Template, Application, and Service Management.

## Use Cases Diagram



*Figure 1 – AIA Use Cases Diagram.*

## SDK Developer Role Use Cases

### Create Template - UC\_APP\_SDK\_SD\_001

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_001 |
| **Title** | Create Template |
| **Description** | The goal of this use case is to allow the actor to create a new template and can download a ZIP archive which contains a standard directory structure as well as a PBA file. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * Supported Technologies (Services) are properly defined in the Service Registry and are linked to some valid docker containers. * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The new template ZIP archive should be created and passed back to the user for download. * This ZIP archive should contain a PBA file with the appropriate data sinks, data sources and data processors definitions. |
| **Main Flow** | * The actor will begin by using the AIA channels (CLI or GUI) to initiate the use case. * The actor will provide a template name, version and description then chooses the data processor technology. * The actor will start adding data sinks and data sources (extension and integration points respectively) and providing template properties values. * The actor will finalize the use case by ‘Saving’ the newly created template from which he/she will receive back a ZIP archive containing the template artifacts. |
| **Alternative Flow (s)** | * The actor should be blocked from ‘Saving’ / ‘Creating’ a template if the mandatory template information is not provided such as name, version, and description…etc. * Data sinks, data sources and data processors are mandatory items to create a new template. |
| **Frequency of use** | Rare. |
| **Priority** | High. |
| **Notes** | N/A. |

### Extend Template - UC\_APP\_SDK\_SD\_002

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_002 |
| **Title** | Extend Template |
| **Description** | The goal of this use case is to allow the actor to create a new template based on another published one. By extending the existing template, the new one will inherit the same data sinks, sources, and processors but with different template name and version. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * At least one published template exists in the platform. * Supported Technologies (Services) are properly defined in the Service Registry and are linked to some valid docker containers. * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The new template ZIP archive should be created and passed back to the actor for download. * This ZIP archive should contain a PBA file with the appropriate data sinks, data sources and data processors definitions. |
| **Main Flow** | * The actor will begin by using the AIA channels (CLI or GUI) to initiate the use case. * The actor will choose a specific template from the list of existing published templates. * The actor will provide a template name, version and description then chooses the data processor technology. * Data sinks and sources will be already populated based on the selected original template which the actor can modify or add new ones. * The actor will finalize the use case by ‘Saving’ the newly created template from which he/she will receive back a ZIP archive containing the template artifacts. |
| **Alternative Flow (s)** | * The actor should be blocked from ‘Saving’ / ‘Creating’ a template if the mandatory template information is not provided such as name, version, and description…etc. * Data sinks, data sources and data processors are mandatory items to create a new template. |
| **Frequency of use** | Rare. |
| **Priority** | High. |
| **Notes** | N/A. |

### Register Template - UC\_APP\_SDK\_SD\_003

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_003 |
| **Title** | Register Template |
| **Description** | The goal of this use case is to allow the actor to publish a new template in AIA Platform by uploading the template ZIP archive. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * Supported Technologies (Services) are properly defined in the Service Registry and are linked to some valid docker containers. * Actor has the sufficient privilege to execute this operation. * Actor has already created the template and obtained the template ZIP archive. |
| **Post-conditions** | * The new template is published into AIA platform and available for the rest of the AIA community to use. * The new template’s ZIP content is stored successfully into Git/Nexus repository. * The new template’s PBA file is registered successfully into the metadata service. |
| **Main Flow** | * The actor will begin by using the AIA channels (CLI or GUI) to initiate the use case. * The actor will upload the template’s ZIP archive. * The system should validate the ZIP contents and start the publishing process and return a descriptive ‘In Progress’ message to the actor. * When finished, the actor should receive a descriptive confirmation or error message. |
| **Alternative Flow (s)** | * The uploaded template ZIP archive is a valid template (both from file type and content point of view), if not then the use case should be blocked and a descriptive error message is sent back to the actor. * Validation rules include:   + PBA structure validation.   + Template name and version.   + Allowed file types in archive for security and data integrity. |
| **Frequency of use** | Rare. |
| **Priority** | High. |
| **Notes** | N/A. |

### Unregister Template - UC\_APP\_SDK\_SD\_004

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_004 |
| **Title** | Unregister Template |
| **Description** | The goal of this use case is to allow the actor to Unpublish an already published template in AIA platform. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * At least one published template exists in the platform. * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The unpublished template is removed completely (including all artifacts in Git/Nexus/Artifactory) and is not available anymore to AIA community. |
| **Main Flow** | * The actor will begin by using the AIA channels (CLI or GUI) to initiate the use case. * The actor will select an already published template. * The actor will confirm unpublishing the template. * The system will unpublish the selected template and confirm back to the actor with a descriptive message. |
| **Alternative Flow (s)** | * The unpublish operation may fail in one of the following validation failure cases:   + Template PBA ID does not exist in metadata service.   + Docker Container does exist for this template’s ID/Version in Artifactory.   + Template ZIP Archive does not exist for this template’s ID/Version in Git/Nexus. |
| **Frequency of use** | Rare. |
| **Priority** | Medium. |
| **Notes** | N/A. |

### List Templates (View Public Catalogue) - UC\_APP\_SDK\_SD\_005

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_005 |
| **Title** | List Templates (View Public Catalogue) |
| **Description** | The goal of this use case is to allow the actor to retrieve the list of available published templates in AIA. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * List of available published templates in AIA platform is retrieved and displayed to the actor. |
| **Main Flow** | * The actor will begin by using the AIA channels (CLI or GUI) to initiate the use case. * The list of available published templates is displayed to the actor including the templates’ names, versions, and descriptions. |
| **Alternative Flow (s)** | * Typically, this use case does not involve any business validation and in case of no templates are published in AIA, an empty list should be returned to the actor. |
| **Frequency of use** | High. |
| **Priority** | High. |
| **Notes** | N/A. |

### Delete Template - UC\_APP\_SDK\_SD\_007

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_007 |
| **Title** | Delete Template |
| **Description** | The goal of this use case is to allow the actor to delete a draft copy of a template from the local machine storage. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The template entry is deleted from the actor’s workspace. |
| **Main Flow** | * The actor will begin by using the AIA channels (Typically, GUI only in this case) to initiate the use case. * The actor navigates to his/her own workspace where all the draft created templates are displayed. * The actor selects a certain template and deletes it. * The system deletes the selected template’s entry from the local machine storage (Currently relies on the browser’s internal HTML 5 based storage) |
| **Alternative Flow (s)** | * Typically, this use case does not involve any business validation and in case of no draft templates are created, an empty list/workspace should be displayed to the actor. |
| **Frequency of use** | Medium. |
| **Priority** | Medium. |
| **Notes** | N/A. |

### Register Service - UC\_APP\_SDK\_SD\_001

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SD\_001 |
| **Title** | Register Service |
| **Description** | The goal of this use case is to allow the actor to define a new technology supported by AIA platform by providing the related PBA and docker images. |
| **Actor(s)** | SDK Developer. |
| **Pre-conditions** | * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The Service is properly defined in AIA platform and available during templates creation. * This means that the service valid PBA file is stored in the metadata store and that a related runnable docker image exists in the Artifactory. * The service dependencies definition should be complete and correct as well. |
| **Main Flow** | * The actor will begin by using the service management tool to initiate this use case. * The actor will provide a valid service PBA definition file, docker image and deployment descriptor artifact (Helm Chart for example – Note: To be confirmed). * The actor will finalize the use case by saving the new service which should add each artifact into metadata store and docker artifactory. |
| **Alternative Flow (s)** | * The system should perform the below validations and fails if any of them did not pass displaying a descriptive error message:   + Verify that the service PBA definition file is valid from structure point of view and that it contains the correct dependencies.   + Verify that the associated docker container is runnable.   + Verify successful storage in the docker artifactory. |
| **Frequency of use** | Rare. |
| **Priority** | High. |
| **Notes** | In future release, a Service Management tool maybe introduced to handle all the CRUD operations related to services/technologies in AIA platform. |

## Application Flow Developer Role Use Cases

### Create Application - UC\_APP\_SDK\_AFD\_001

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_AFD\_001 |
| **Title** | Create Application |
| **Description** | The goal of this use case is to allow the actor to create a new AIA application based on an existing published template. |
| **Actor(s)** | Application Flow Developer. |
| **Pre-conditions** | * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The actor receives a ZIP archive for the new application containing the related PBA file according to the actor’s choices while building the application. * No associated artifacts are created either in the metadata service, Git/Nexus or artifactory. |
| **Main Flow** | * The actor will begin by using the AIA Portal to initiate the use case. * The actor will begin by providing a new application’s name, description and selects a template. The system should suggest a version number as a smart default. * The actor will add the application’s extension and integration points either from the published applications catalogue or from the selected template in the above step. * The actor will provide the extension/integration points’ attributes values (Example: URI, driver, password, topic…etc) * The actor will finalize the operation by saving the newly created application and the system should respond back with a downloadable ZIP file containing the new application PBA and structure. |
| **Alternative Flow (s)** | * The system should perform the following validations and if any of them failed, then the use case should be blocked and a descriptive error message should be returned to the actor:   + Verify that all the mandatory input fields values are present. |
| **Frequency of use** | High. |
| **Priority** | High. |
| **Notes** | N/A. |

### Extend Application - UC\_APP\_SDK\_AFD\_002

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_AFD\_002 |
| **Title** | Extend Application |
| **Description** | The goal of this use case is to allow the actor to create a new AIA analytics application based on an existing one by extending it. |
| **Actor(s)** | Application Flow Developer. |
| **Pre-conditions** | * At least one published AIA application exists in the application catalogue. * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The new application ZIP archive should be created and passed back to the actor for download. * This ZIP archive should contain a PBA file with the appropriate data sinks, data sources and data processors definitions. |
| **Main Flow** | * The actor will begin by using one of the AIA Portal to initiate the use case. * The actor will select an existing published application from the application catalogue and chooses to extend it. * The system will redirect the actor to the same steps as when creating a new application but the related fields will be populated with the original’s application data. * The actor will provide the new data sources and sinks configurations. * The actor will finalize the use case by saving the new application. * The system should respond back with the new extended application’s ZIP archive to be downloaded by the actor. |
| **Alternative Flow (s)** | * The actor should be blocked from ‘Saving’ / ‘Creating’ an application if the mandatory application information is not provided such as name, version, and description…etc. * Data sinks, data sources and data processors are mandatory items in order to create a new application. |
| **Frequency of use** | High. |
| **Priority** | High. |
| **Notes** | N/A. |

### Unpublish Application - UC\_APP\_SDK\_AFD\_003

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_AFD\_003 |
| **Title** | Unpublish Application |
| **Description** | The goal of this use case is to allow the actor to unpublish an already published application. |
| **Actor(s)** | Application Flow Developer. |
| **Pre-conditions** | * At least one published application exists in the applications catalogue. * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * The selected application is completely removed from the applications catalogue – along with is artifacts – and is no longer available to the AIA community for re-use. |
| **Main Flow** | * The actor will begin by using the AIA Portal to initiate the use case. * The actor will select a specific published application (by name and version). * The actor will choose to unpublish this application. * The system will perform the unpublishing operation by marking the application PBA as inactive in the metadata service and deleting the associated docker artifacts and ZIP archives in Git/Nexus and artifactory. * The system should respond back to the actor with a descriptive confirmation message. |
| **Alternative Flow (s)** | * The unpublish operation may fail in one of the following validation failure cases:   + Application PBA ID does not exist in metadata service.   + Docker Container does exist for this application’s ID/Version in Artifactory.   + Application ZIP Archive does not exist for this application’s ID/Version in Git/Nexus. |
| **Frequency of use** | Medium. |
| **Priority** | High. |
| **Notes** | N/A. |

### Publish Application - UC\_APP\_SDK\_AFD\_005

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_AFD\_005 |
| **Title** | Publish Application |
| **Description** | The goal of this use case is to allow the actor to publish a new application and make it publicly available to AIA community. |
| **Actor(s)** | Application Flow Developer. |
| **Pre-conditions** | * Supported Technologies (Services) are properly defined in the Service Registry and are linked to some valid docker containers and the same is true for the template used by the application. * Actor has already created the application and obtained its ZIP archive. * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** |  |
| **Main Flow** | * The actor will begin by using the AIA channels (CLI or GUI) to initiate the use case. * The actor will upload the application’s ZIP archive. * The system should validate the ZIP contents and start the publishing process and return a descriptive ‘In Progress’ message to the actor. * When finished, the actor should receive a descriptive confirmation or error message. |
| **Alternative Flow (s)** | * The uploaded application ZIP archive is a valid application (both from file type and content point of view), if not then the use case should be blocked and a descriptive error message is sent back to the actor. * Validation rules include:   + PBA structure validation.   + Application name and version.   + Allowed file types in archive for security and data integrity. |
| **Frequency of use** | Medium. |
| **Priority** | High. |
| **Notes** | N/A. |

### List Applications (View/Search Public Catalogue) - UC\_APP\_SDK\_AFD\_006

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_AFD\_006 |
| **Title** | List Applications (View/Search Public Catalogue) |
| **Description** | The goal of this use case is to allow the actor to retrieve the list of available published application in AIA platform. |
| **Actor(s)** | Application Flow Developer. |
| **Pre-conditions** | * Actor has the sufficient privilege to execute this operation. |
| **Post-conditions** | * List of available published applications in AIA platform is retrieved and displayed to the actor based on the indicated criteria. |
| **Main Flow** | * The actor will begin by using the AIA Portal to initiate the use case. * The list of available published applications is displayed to the actor including the applications’ names, versions, and descriptions. * The actor can also use a search box in the GUI to search for published applications by name. |
| **Alternative Flow (s)** | * Typically, this use case does not involve any business validation and in case of no applications are published in AIA, an empty list should be returned to the actor. |
| **Frequency of use** | High. |
| **Priority** | High. |
| **Notes** | N/A. |

## Solution Engineer Role Use Cases

### Compose Application - UC\_APP\_SDK\_SE\_001

|  |  |
| --- | --- |
| **ID** | UC\_APP\_SDK\_SE\_001 |
| **Title** | Compose Application. |
| **Description** | The goal of this use case is to allow the actor to compose a new logical application based on mixing and matching existing microservices (flows) inside published applications. |
| **Actor(s)** | Solution Engineer. |
| **Pre-conditions** | * Actor has the sufficient privilege to execute this operation. * One more published applications at least exist in the application catalogue. |
| **Post-conditions** | * The actor receives a ZIP archive for the new application containing the related PBA file according to the actor’s choices while building the application. * No associated artifacts are created either in the metadata service, Git/Nexus or artifactory. |
| **Main Flow** | * The actor will begin by using the AIA Portal to initiate the use case. * The actor will begin by providing a new application’s name, description and selects the desired flows (microservices) from the list of existing published application’s internal flows. * The actor will customize the new application’s extension and integration points and flows configuration (input/output). * The actor will provide the extension/integration points’ attributes values (Example: URI, driver, password, topic…etc) * The actor will finalize the operation by saving the newly created application and the system should respond back with a downloadable ZIP file containing the new application PBA and structure. |
| **Alternative Flow (s)** | * The system should perform the following validations and if any of them failed, then the use case should be blocked and a descriptive error message should be returned to the actor:   + Verify that all the mandatory input fields values are present.   + Verify that the flows (microservices) configuration are technically valid. |
| **Frequency of use** | Medium. |
| **Priority** | High. |
| **Notes** | \*\* This use case is still in development and the above description is an early draft as this is an advanced use case which will be included in later stages. |

# Appendix A – Open Issues

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Issue Description** | **Status** | **Related JIRA Ticket #** |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |