**Identificador**

**Nombre Decisión**

**Contexto**

**Requisitos**

**Descripción**

**Argumentación**

**Alternativas**

**Pros y Cons**

{ADD-001}

* Nombre Decisión: [Estilo por Eventos]
* Estatus: [Propuesto]
* Fecha: [2022-11-08]

**Requisitos**

\*[RF-02] \*[RF-03] \*[RF-09]

**Descripción**

{Está compuesta por productores y consumidores de eventos. Una vez que se detecta un evento, este se transmite a una plataforma donde se procesa de manera asíncrona para procesar los eventos. La plataforma de procesamiento ejecutará la respuesta adecuada para el evento y enviará la actividad a los consumidores correspondientes.}

**Alternativas**

* {Estilo por Capas}
* {Estilo Pipe and Filter}

**Argumentación**

{}

**Consecuencias Positivas**

**Consecuencias Negativas**

{ADD-002}

* Nombre Decisión: [Factory Method]
* Estatus: [Propuesto]
* Fecha: [2022-11-08]

**Requisitos**

\*[RF-05]

**Descripción**

{Es un patrón de diseño creacional que proporciona una interfaz para crear objetos en una superclase, mientras permite a las subclases alterar el tipo de objetos que se crearán.}

**Alternativas**

* {Abstract Factory}

**Argumentación**

{}

**Consecuencias Positivas**

**Consecuencias Negativas**

{ADD-003}

* Nombre Decisión: [Observer]
* Estatus: [Propuesto]
* Fecha: [2022-11-08]

**Requisitos**

\*[RF-06] \*[RF-07]

**Descripción**

{Es un patrón de diseño de comportamiento que te permite definir un mecanismo de suscripción para notificar a varios objetos sobre cualquier evento que le suceda al objeto que están observando.}

**Alternativas**

* {Publish-Suscribe}

**Argumentación**

{}

**Consecuencias Positivas**

**Consecuencias Negativas**

{ADD-004}

* Nombre Decisión: [Estilo por Capas]
* Estatus: [Propuesto]
* Fecha: [2022-11-08]

**Requisitos**

\*[RF-06] \*[RF-07]

**Descripción**

{Todas las capas se colocan de forma horizontal, de tal forma que cada capa solo puede comunicarse con la capa que está inmediatamente por debajo, por lo que, si una capa quiere comunicarse con otras que están mucho más abajo, tendrán que hacerlo mediante la capa que está inmediatamente por debajo}

**Alternativas**

* {Estilo Pipe and Filter}

**Argumentación**

{}

**Consecuencias Positivas**

**Consecuencias Negativas**

Estilo Principal.md

* Estatus: [propuesto]
* : [ASS Grupo 1]
* Fecha: [2022-11-08]

**Contexto del problema**

**Requisitos**

[RF-02], [RF-09], [RF-03]

**Opciones Consideradas**

* Estilo por Capas
* Estilo Pipe and Filter
* Estilo por Eventos
* Estilo por Microservicios

**Decisión tomada**

**Consecuencias Positivas**

**Consecuencias Negativas**

Título

Estatus: Pendiente

Identificador: ADD-005.md

Problema

Existe una familia de dispositivos IoT compuesta por tres sensores en los que el primero envía información al segundo y este al tercero que finalmente lo envía al centro de notificaciones.

Requisitos

RF-08

Opciones

Estilo por capas

Estilo Pipe and Filter

Decisión Tomada

Estilo por capas

Consecuencias Positivas

Consecuencias Negativas

Identificador

ADD-004

Problema

Existen tres familias de sensores, cada una de las cuales comparte cierta funcionalidad, pero dispone de otras diferentes entre una familia y otra.

Requisitos

RF-05

Decisión

Factory Method

Estatus

Pendiente

Alternativas

Abstract Factory

Consecuencias Positivas

Consecuencias Negativas

Identificador

ADD-007

Problema

Además, los operarios de la factoria 4.0 debe estar permanente notificados a través de un sistema de mensajería interno y deben poderse suscribir a diferentes eventos y notificaciones como actualizaciones de la producción, fallos en los sensores o sobrecarga en la producción.

Requisitos

RF-06, RF-07

Decisión

Observer

Estatus

Pendiente

Alternativas

Publish-Suscribe

Consecuencias Positivas

Consecuencias Negativas