



**UNIVERSIDAD  
DE ANTIOQUIA**

**Metodología de Gestión de Riesgos  
Aplicación Contexto**

**Autor(es)**

**Alejandro Isaza Delgado**

**Universidad de Antioquia**

**Facultad Ingeniería, Departamento Sistemas**

**Medellín, Colombia**

**2020**



Metodología de Gestión de Riesgos

Aplicación Contexto

**Alejandro Isaza Delgado**

Informe practica empresarial presentada(o) como requisito parcial para optar al título de:

**Pregrado en Ingeniería de Sistemas**

Asesores (a):

Roberto Florez Rueda, Ingeniero de Sistemas

Johana Paola Zuleta Torres, Ingeniera informática

Universidad de Antioquia

Facultad Ingeniería, Departamento Sistemas

Medellín, Colombia

2020

## **Resumen**

Se necesita una aplicación en la cual se pueda centralizar la información relacionada al contexto de una empresa afiliada ya que para el manejo de esta información de la empresa se manejan diferentes formatos y se usan diferentes fuentes de información lo que conlleva más esfuerzo y tiempo para los asesores y muchas veces información duplicada y/o falta de información a tener en cuenta para estar bien informados sobre alguna empresa y su contexto.

La aplicación contexto es un sistema en el cual los asesores de la empresa pueden agregar y ver la información de una empresa afiliada, en estos se incluye información empresarial, información de su relacionamiento y algunos resultados y variables relacionadas con las empresa y de esta manera tener la información que actualmente tiene ya los asesores en otros formatos y otras aplicaciones de la empresa en una sola aplicación con información centralizada, la cual contribuirá a un mejor establecimiento del contexto de la empresa dentro la gestión de riesgos y de esta forma poder darle un mejor servicio por parte de la empresa que ofrece el servicio de gestión de riesgos y sus asesores a las empresas afiliadas.

**Palabras claves:** Sistema gestión de riesgos, Scrum, Contexto, Event Storming, DDD, Arquitectura hexagonal.

## **Introducción**

Dentro del proceso de gestión de riesgos en la empresa prestadora del servicio se requiere una aplicación la cual cumpla con la función de ayudar a las personas alrededor de una empresa cliente a establecer el contexto de esta empresa de una forma unificada y fácil de usar, ya que hoy en día se establece este contexto mediante plantillas las cuales se vuelven tediosas de unificar por parte de los asesores y no se mantiene una información unificada. para esto se desarrolla una aplicación para la empresa que presta el servicio de gestión de riesgos el cual tenga como mínimo producto viable estas características: consultar los datos de una empresa desde las diferentes fuentes de información que se poseen actualmente notificando los cambios que se realizan sobre estos a un asesor que tenga acceso a esta información; poder actualizar la información de una empresa notificando los cambios a un asesor; y generar un informe con los datos de contexto de una empresa el cual sirva como insumo para los asesores, en esta aplicación se considera como contexto de una empresa la información empresarias, los datos de la empresa en relación con la empresa prestadora del servicio y información de resultados y variables de la empresa en la gestión de riesgos. la metodología empleada para la construcción de la aplicación fue mediante ciclos incrementales de 15 días en los cuales se usó el marco de trabajo scrum, una arquitectura limpia como lo es la arquitectura hexagonal y buenas prácticas de desarrollo de software.

## **Objetivos**

### **Objetivo general**

Facilitar y ayudar a mejorar el proceso por el cual se establece el contexto de una empresa

### **Objetivos específicos:**

- Conocer el cliente, sus objetivos estratégicos y el entorno (interno y externo) en que se mueve.
- Conocer no sólo las desventajas sino también las oportunidades.
- Garantizar la pertinencia de la evaluación y el tratamiento de riesgo y alinearlos con los objetivos del cliente.
- Saber de antemano qué se ha hecho con el cliente y por qué.
- Establecer marcos de referencia y métodos de identificación a utilizar.

## Marco Teórico

### Proceso de Gestión de riesgos

la gestión de riesgos dentro de una compañía es muy importante ya que esta ayuda a la mejora continua de los procesos de las empresas y para una adecuada gestión de riesgos la Organización Internacional de Normalización (iso) define la norma iso-31000 la cual nos proporciona pautas sobre la gestión de riesgos que enfrentan las organizaciones [1] y dentro de este proceso se identifican 5 etapas importante de la gestión de riesgos dentro de las cuales está el establecimiento del contexto la cual es la etapa que se va a tratar sobre proyecto y la cual ayudará a mejorar con la aplicación.

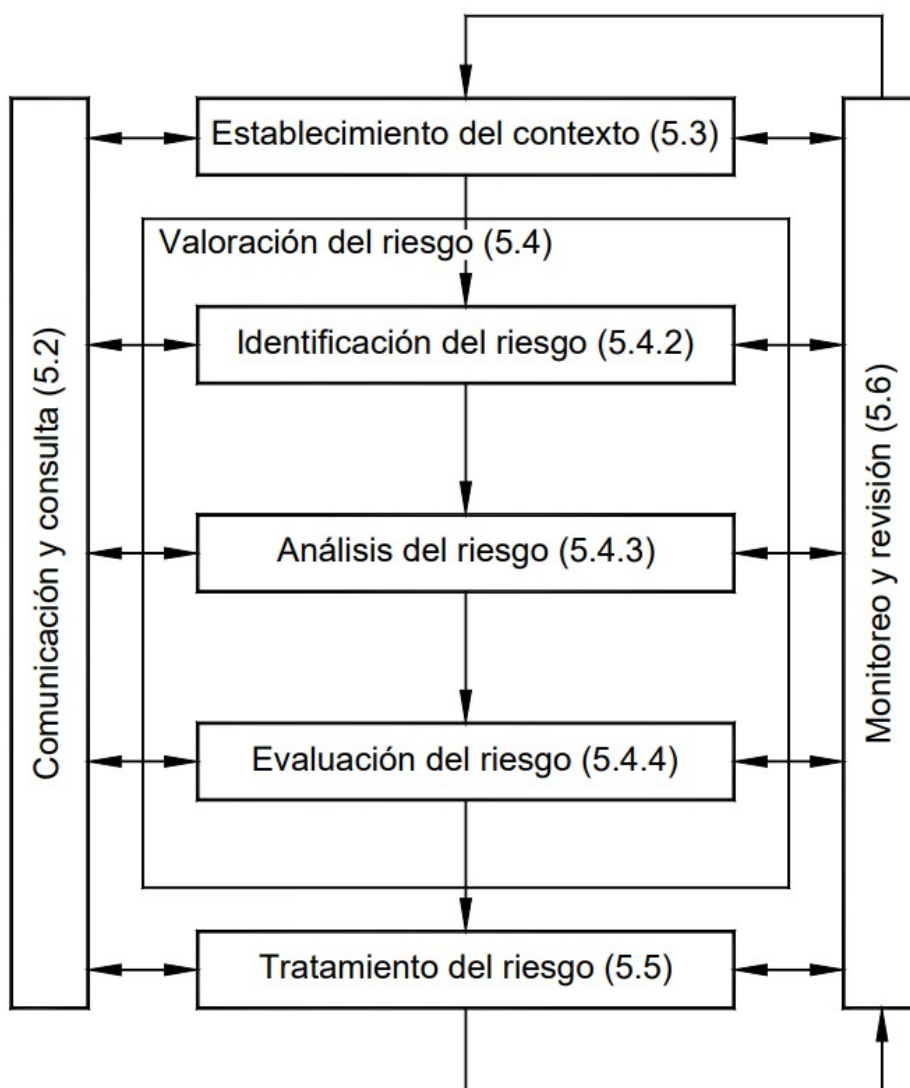


Figura 1: proceso para la gestión de riesgos [1]

## **Contexto**

Entendimiento del cliente, su realidad, estrategia, metas, objetivos, capacidades, necesidades, estructura y prioridades.

- Definición de los parámetros internos y externos que se han de tomar en consideración cuando se gestiona el riesgo
- Conocer el cliente, sus objetivos estratégicos y el entorno en que se mueve
- Conocer no sólo los riesgos sino también las oportunidades
- Garantizar la pertinencia de la evaluación y el tratamiento de riesgo y alinearlos con los objetivos del cliente
- Qué se ha hecho con el cliente y por qué?
- Apetito de riesgo
- Impulsores clave y tendencias que pueden tener impacto en los objetivos de la organización
- “There must first be objectives before management can identify and assess risks to their achievement and necessary actions to manage the risk” – Airmic 2005, Risk Management Standard

## Contexto en la práctica

- Información básica/general del cliente
  - Identificación del cliente
  - Sector económico
  - Objetivos estratégicos
  - Segmentación
  - Contactos / Estructura de relacionamiento
  - Estructura organizacional, procesos
  - Apetito de riesgo
- Riesgos/Diagnósticos más relevantes
- Resumen plan de trabajo con el cliente
- Evolución en Gestión de Riesgo
- Relaciones con partes involucradas

## Tecnologías

Para el desarrollo del proyecto se tuvieron en cuenta las mejores tecnologías para lo que se necesita tecnologías que ayuden a la escalabilidad del proyecto y al desarrollo de este mismo teniendo en cuenta que este hace parte de un ecosistema de microservicios. Las tecnologías usadas fueron las siguientes:

### **Angular:**

Angular es un framework de aplicaciones y una plataforma de desarrollo para crear aplicaciones eficientes y sofisticadas de una sola página [6]. se hace uso de angular ya que es un framework muy robusto que que tiene un marco de trabajo muy bien estructurado lo cual es de mucha ayuda para la escalabilidad del proyecto y el fácil entendimiento del código.



**Play Framework:**

Play se basa en una arquitectura ligera, sin estado y amigable para la web. Basado en Akka, Play proporciona un consumo de recursos predecible y mínimo (CPU, memoria, subprocesos) para aplicaciones altamente escalables. [8], se hizo uso de este framework de ya que usa java el cual es un lenguaje muy robusto y es el más usado para aplicaciones empresariales por esta misma razón, también es un framework que ayuda mucho a la reactividad y el desarrollo de aplicaciones que no usen muchos recursos del sistema.

**Oracle:**

Oracle es un gigante en el ámbito de las tecnologías mundial y es el sistema de bases de datos que usa la empresa que presta el servicio ya que es un sistemas muy poderoso para manejar grandes cantidades de información y poder procesarlos de una forma rápida. Oracle no es precisamente la mejor base de datos para un sistema reactivo que pero en este caso como es la base de datos predilecta de la empresa y nos ofrece mucha robustez se opta por hacer uso de esta.

**Jenkins:**

Jenkins, el principal servidor de automatización de código abierto, proporciona cientos de complementos para respaldar la creación, implementación y automatización de cualquier proyecto [9]. jenkins nos ayuda en toda la automatización del proyecto y de una forma muy fácil poder realizar pruebas de integración y de rendimiento automatizadas y realizar despliegues en todos los ambientes.

## **Event Storming**

Event Storming es una exploración y modelado colaborativo de workshops complejos de dominios de negocio. Involucra la unión de todos los stakeholders de un proyecto para alinearse en una comprensión tecnológicamente-neutral del dominio de negocio y del problema acuciante. Esto sitúa la solución en el contexto de negocio apropiado, ayudando a asegurar que los expertos en negocios y los expertos en tecnología lleguen a un entendimiento antes de construir un sistema [5].

## **Domain driven design (DDD)**

En el desarrollo de este proyecto también se hace uso de domain driven design ya que es un enfoque para el desarrollo de software que nos propone pensar de una forma un poco diferente a la hora del diseño y del desarrollo para implementar o usar conceptos más relacionados al negocio al cual va a usar el sistema.

La mejor forma de entender los conceptos de este esquema de desarrollo, es olvidar por un momento todo lo que conoces sobre programación y pensar en el mundo real. La OOP ha sido abstraída del mundo real y, por tanto, es una representación de este. Del mismo modo, el DDD es una abstracción de cómo hemos estructurado el mundo real y buscar similitudes entre ambos te facilitará la tarea de entenderlo.

DDD no define implementaciones, define conceptos y 'reglas de implementación'. Por tanto es independiente de cualquier framework y lenguaje [6].

## **Command Query Responsibility Segregation (CQRS)**

CQRS es un patrón que lo que nos propone es separar el modelo de datos de la información que se guarda en nuestra sistema del modelo de datos para consultar esta información, se parar el modelo en consultar y comandó, las consultas son la información que se consulta y los comando la información que se actualizan, este patrón nos ayuda mucho para escalar de forma individual cada modelo por aparte en caso de que se use mucho más un modelo que el otro [10].

En este proyecto de contexto también usamos cqrs pero de una forma un poco diferente más adaptada a nuestras necesidades y también de una forma más básica ya que todavía no se a podido por ejemplo usar los modelos en sistemas separados o usar diferentes bases de datos, pero lo que se tiene actualmente sirve como base para en un futuro si se necesita de forma mucho más fácil escalar la aplicación.

# Event Storming Contexto

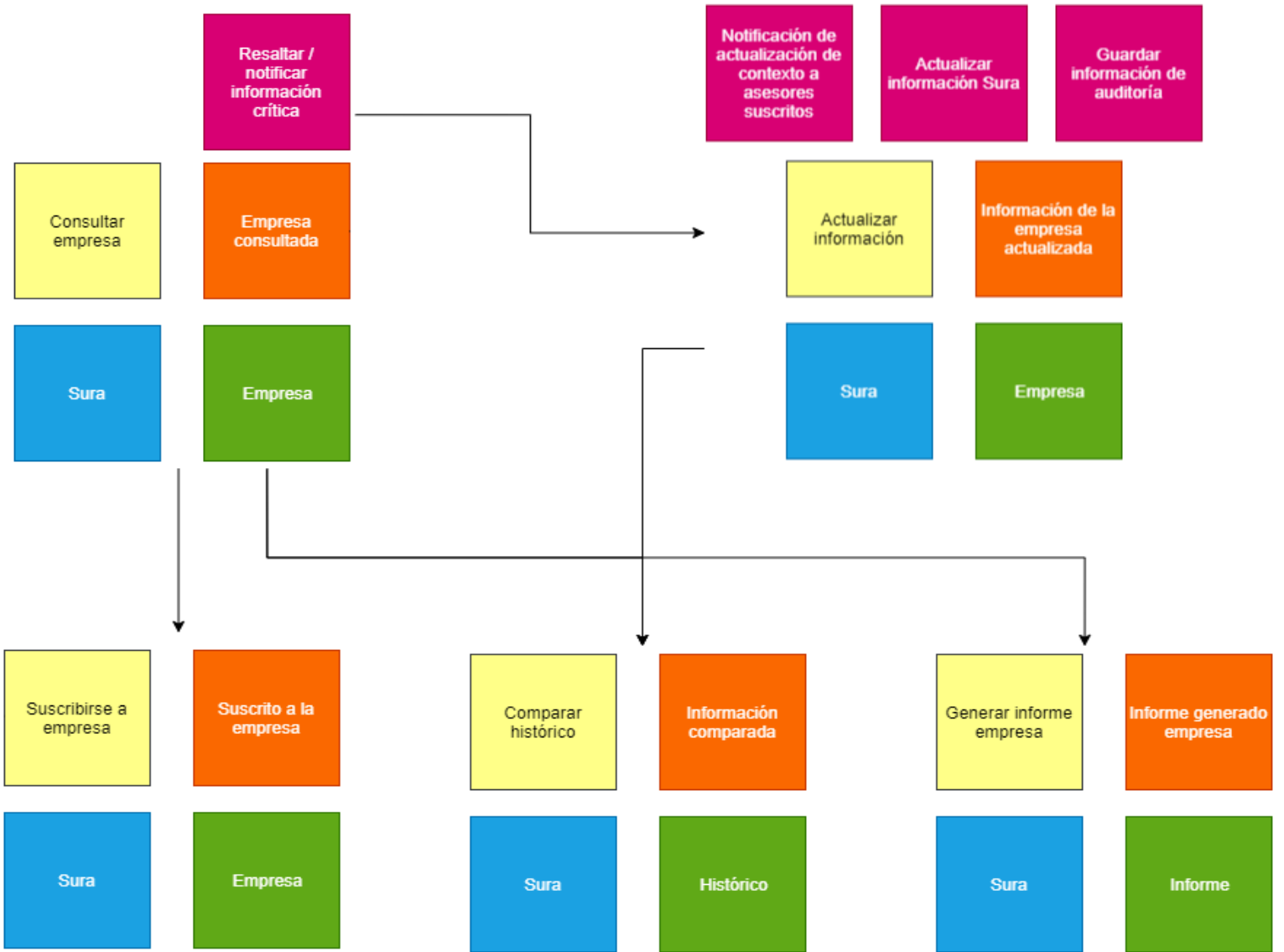
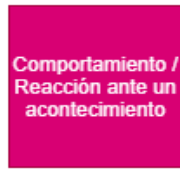
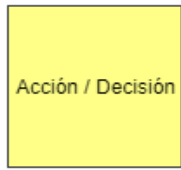
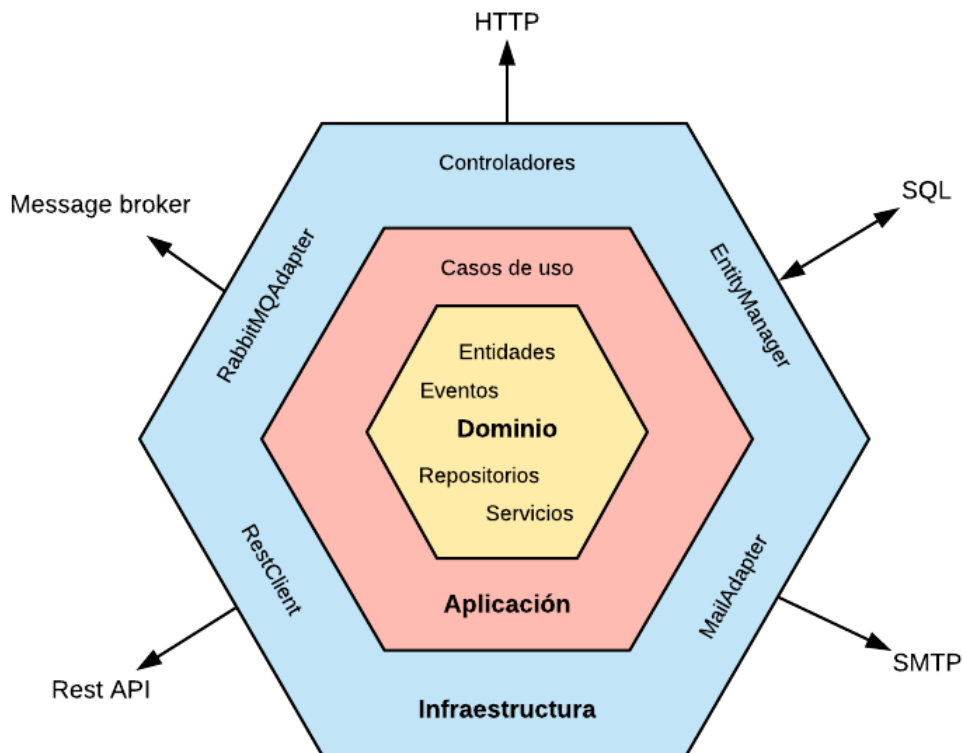


Figura 2: Event Storming Aplicación Contexto.

## Arquitectura

Para el desarrollo de la aplicación se usará arquitectura hexagonal [3] más conocida como arquitectura de puertos y adaptadores, y microservicios. [4] esto nos ayudará mucho a la escalabilidad y mantenimiento de la aplicación ya que esta separa el software en tres capas: la capa del dominio será la capa interna y en esta se encuentra toda la lógica del negocio y los modelos de este, la capa media es la capa de aplicación es la cual se encarga de tener los casos de uso y como tal los pasos de casa proceso de la aplicación y la capa externa es la capa de infraestructura en la cual se manejan los puntos de entrada y salida a la aplicación y lo que sería externo a esta.



**Figura 3: arquitectura hexagonal.**

## **Metodología**

Este proyecto se realizará mediante marco de trabajo ágil scrum y buenas prácticas de desarrollo de software. Para lo cual se deberán seguir los siguientes pasos:

1. Reuniones con el PO por parte de la empresa, quien oficiara como experto en el negocio y usuario del aplicativo. En estas reuniones se realizaran revisiones del historias de usuario y seguimiento del desarrollo del producto. (review, refinamiento y planning)
2. El PO se encarga de traducir las necesidades del negocio en historias de usuario para poder desarrollar la aplicación.
3. Se realiza el desarrollo de la aplicación teniendo en cuenta la historias de usuario y buenas prácticas mediante sprints de dos semanas.
4. Se realizan pruebas con el PO.
5. Aprobación, entrega y salida a producción de lo desarrollado en el sprint.

## **Resultados y análisis**

Se logra terminar la primera versión de esta aplicación con el producto mínimo viable aunque no se hizo lo de notificar a los asesores por cambios en la planeación del proyecto y por que se tenían otras prioridades pero igual la aplicación logra lo que se propuso de ayudar a los asesores a realizar el contexto de una empresa y actualmente se le está dando uso y han tenido una buena impresión de la aplicación, ya que efectivamente desde esta pueden tener centralizada la información relacionada con el contexto de una empresa, también con el informe que se saca desde la aplicación los asesores tienen un forma fácil de retroalimentar a las empresas clientes sobre cómo está su contexto sus variables a tener en cuenta a la hora de realizar el proceso de gestión de riesgos. También nos hacen feedback sobre lo que se debería cambiar y errores que encuentran en la aplicación para tener en cuenta en futuras versión de esta y en mejoras continuas. hoy en dia estamos promediando un uso de más o menos mil asesores de la empresa prestadora del servicio que usan la aplicación cada mes, este número no es muy grande dentro del total de asesoras de la empresa pero es importante para la aplicación como tal y poder darla a conocer y que en el futuro pueda ser más usada y más aprovechada.

## **Conclusiones**

Como se menciona el los resultados se logró construir una aplicación con todas tecnologías mencionadas anteriormente de forma exitosa y todas estas fueron buena decisiones ya que nos aportan mucho a los que es el comportamiento de la aplicación y a la escalabilidad.

La aplicación se encuentra actualmente en pdn y ya está dando buenos resultados y ahorrándole tiempo a los asesores de la empresa, la metodología elegida también nos ha dado muy buen resultado ya que se puede salir a producción en cada sprint de esta forma tener en producción los más recientes cambios y mejoras que se le fueron haciendo a la aplicación y también gracias a la automatización de las pruebas y el despliegue. También fue muy importante las reuniones que se hacen usualmente en sprint poderlas hacer virtualmente ya que apesar de el estado actual en que se encuentra la sociedad colombiana y mundial con respecto a la pandemia que se viene viviendo por estos días no nos afectó el normal desarrollo del proyecto y se pudo seguir avanzando con el normal cronograma que se planteó sobre este.

Con respecto a los objetivos planteados si se cumplieron ya que la aplicación cumple con facilitarle la vida a los asesores en el proceso de establecer el contexto y conocer mejor al cliente para darle un mejor servicio.

## Referencias bibliográficas

[1] NORMA TÉCNICA COLOMBIANA, GESTIÓN DEL RIESGO. PRINCIPIOS Y DIRECTRICES. 2011. [https://sitios.ces.edu.co/Documentos/NTC-ISO31000\\_Gestion\\_del\\_riesgo.pdf](https://sitios.ces.edu.co/Documentos/NTC-ISO31000_Gestion_del_riesgo.pdf)

[2] ¿Qué es la gestión de riesgos en las empresas?.  
<http://herramientas.camaramedellin.com.co/Inicio/Buenaspracticasesempresariales/BibliotecaJur%C3%ADdica/%C2%BFQueeslagestionderiesgosenlasempresas.aspx>

[3] Hexagonal Architecture. <https://fideloper.com/hexagonal-architecture>

[4] A pattern language for microservices. <https://microservices.io/patterns/index.html>

[5] Event Storming: Ventajas y Técnicas. <https://apiumacademy.com/es/blog/event-storming-ventajas-y-tecnicas/>

[6] Resumen sobre DDD. Domain Driven Design.  
[https://github.com/jatubio/5minutos\\_laravel/wiki/Resumen-sobre-DDD.-Domain-Driven-Design](https://github.com/jatubio/5minutos_laravel/wiki/Resumen-sobre-DDD.-Domain-Driven-Design)

[7] Introduction to the Angular Docs]. <https://angular.io/docs>

[8] Play Framework makes it easy to build web applications with Java & Scala.  
<https://www.playframework.com/>

[9] Jenkins Build great things at any scale. <https://www.jenkins.io/>

[10] CQRS. <https://martinfowler.com/bliki/CQRS.html>