



UNIVERSIDAD DE
COSTA RICA

SG-CIE

Carrera de
Informática Empresarial
Sede de Guanacaste

Curso de Gestión de Proyecto IF-7201

Investigación

SaFe

Estudiante

Vasquez Murillo Erick - B98334

Docente:
Hazel Bustos Muñoz

Liberia
Junio de 2022

índice

Preguntas.....	3
1. Tabla comparativa con los Roles de equipo SAFe.....	3
2. Roles de proceso SAFe: Nivel de programa	4
3. Marco de trabajo de SAFe ¿qué es un tren?.....	5
4. “SAFe es una metodología de desarrollo de software”.....	5
5. Diferencia entre un Spike en SAFe y una caja de tiempo en Scrum	6
6. Defina los siguientes elementos de trabajo en SAFe y de un ejemplo	6
a. Epic de programa.....	6
b. Característica Historia.....	7
c. Objetivo de PI.....	7
d. Tarea.....	7
e. Defecto.....	7
f. Riesgo.....	8
g. Retrospectiva	8

Preguntas

1. Tabla comparativa con los Roles de equipo SAFe

La siguiente tabla comparativa, son los roles de SAFe a nivel Programa; comparado con los roles scrum. Hay que mencionar que cada rol se divide en diferentes niveles, los cuales son: Equipo, Programa, Solución y Portafolio.

SAFe	Scrum	Descripción
Product Manager	Product Owner	En el nivel programa, el Product Manager hace funciones similares al PO de Scrum. Es importante aclarar que en SAFe, cada equipo cuenta con un Scrum Master y un PO
Release Train Engineer	Scrum Master	En el nivel de Programa, el Release Train Engineer es el líder del Scrum Master
Business Owner	Stakeholders	El Business Owner a nivel de negocio es el conjunto de interesados que en Scrum serian los Stakerholder
System Architect/Engineer	Scrum Team	El System Architect se encargan de definir la arquitectura sistema y otras funciones. El Scrum Team desarrolla distintas funciones según el equipo de Scrum Team, como software, QA, etc.
Customer	Customer	En ambos, son los clientes, finales.

A continuación, se explica cada rol de Scrum:

Scrum Team: es el equipo de desarrollo, común mente es un equipo de software, pero se puede adaptar para otras áreas de TI. La cantidad de miembros es de 5 a 10 miembros.

Scrum Master: es el encargado de liderar al equipo, ayudándole a cumplir con los procesos de Scrum. De igual forma, trabaja junto al Product Owner.

Product Owner: representa a los clientes que van a utilizar el software, se encarga de trasladar la visión del proyecto al equipo. Igual forma, realiza el Product Backlog.

Stakeholders: son las personas interesadas del producto de software, ya sean inversionistas, dueño del producto, etc.

2. Roles de proceso SAFe: Nivel de programa

Product Manager: se encarga de priorizar los requerimientos y debe determinar que estén bien descritas. Se comunica con el Business Owner y con los clientes.

System Architect/Engineer: es un equipo pequeño que se encarga de desarrollar el release train. Realiza actividades como la arquitectura del sistema, realizan los requisitos no funcionales.

Release Train Engineer: es el encargado de liderar al equipo para que el tren de lanzamiento (release train) salga según la metodología. Es el jefe del Scrum Master de cada equipo.

Business Owner: son las partes interesadas del producto (stakeholders). Son los responsables de identificar las necesidades del producto y el retorno de inversión.

Customer: son los clientes finales, estos prueban los resultados finales de cada release train.

3. Marco de trabajo de SAFe ¿qué es un tren?

El Agile Release Train (ART) es una sección de la metodología SAFe donde se desarrolla y entrega soluciones del producto a desarrollar para el usuario final. En comparación de Scrum, son como las iteraciones donde se desarrollan las historias. Se despliega de manera incremental. Al ser una metodología que se aplica a grandes organizaciones, el número de actores que actúan en el ART es entre 50 a 120 personas.

Se desarrolla por medio 3 secuencias, las cuales son: Exploración continua, Integración Continua y Despliegue Continuo. Todos los equipos por desarrollar deben realizar los siguientes pasos: Definir, Desarrollar, Validar y Entregar.

De igual forma dentro de estas secuencias, se encuentran 9 tipos de equipos en donde trabajarán los distintos equipos: Business, Product Management, Hardware, Software, Quality, Testing, Compliance, Operations, Security. Cada equipo es auto-organizado y auto-gestionado.

Dentro de la ejecución de un tren, pueden entregar un flujo de valor pequeño el cual se puede desarrollar en un ART o en un flujo de valor mayor donde deben ser ejecutado con varios ART.

4. “SAFe es una metodología de desarrollo de software”

Esta frase no es del todo cierta, dado que SAFe es una metodología ágil que es aplicada en la organización de toda una empresa (comúnmente en empresas grandes) y el desarrollo de programas de TI (no necesariamente solo software), por lo que el enfoque no es únicamente a los equipos como es en Scrum. Por lo que la frase no es cierta, dado

que no es una metodología que se enfoca en el desarrollo de software, si no en la organización ágil de toda una empresa, la cual esta engloba el desarrollo de software.

La metodología es tan grande que dentro de ella existen roles de Scrum Master para su funcionamiento.

5. Diferencia entre un Spike en SAFe y una caja de tiempo en Scrum

Sipke	Caja de tiempo
técnica que evitar la incertidumbre y riesgo de un evento	técnica para limitar la duración de todos los eventos
estima la confiabilidad de una estimación de la historia	estima el tiempo de una historia
representa actividades como: investigación, diseño, exploración y creación de prototipos.	Representa actividades como: sprint, sprint planing, daily scrum, sprint review, sprint retrospective.
tienen un tamaño máximo de una caja de tiempo	Definen el tiempo de la caja de tiempo
Se divide en spike funcionales y sipke técnicos	Se divide en bloque de tiempo rígido y bloque de tiempo flexible

6. Defina los siguientes elementos de trabajo en SAFe y de un ejemplo

a. Epic de programa

Definición: son grandes cantidades de trabajos que se pueden desglosar en numero de tareas mas pequeños.

Ejemplo: El Testing es un trabajo épico, porque se puede dividir en tareas específicas como testear la interfaz de usuario, testear la base de datos, etc.

b. Característica Historia

Definición: son breves requisitos en un lenguaje natural, solicitado por el representante del cliente.

Ejemplo: una historia es que el usuario desea buscar restaurantes, dentro una la aplicación de comida

c. Objetivo de PI

Definición: es un periodo de tiempo durante el cual el ART entrega un incremento de valor. Los objetivos son: facilitar la funcionalidad, permitir ajustes en recursos y presupuestos, alinear la visión con los entregables, facilitar la supervisión.

Ejemplo: ayuda a planear los objetivos y riesgos, por lo que se trabaja de manera conjunta en el PI Planing.

d. Tarea

Definición: es una asignación específica a realizar, normalmente las tareas son los resultados de dividir las historias.

Ejemplo: La historia de buscar restaurante, se puede dividir en tareas como: crear el input de buscar, programar la lógica para buscar, mostrar los restaurantes que se buscó.

e. Defecto

Definición: es alguna deficiencia con la que cuenta algún sistema u organización dentro de la metodología.

Ejemplo: un defecto puede ser que alguna función de un software no haya salido como lo esperado.

f. Riesgo

Definición: es cualquier evento que puede tener un impacto negativo o positivo en la organización.

Ejemplo: un riesgo común en una organización es la rotación constante del personal.

g. Retrospectiva

Definición: es un evento en el que se realiza feedback de lo realizado en una iteración de proyecto.

Ejemplo: se pueden realizar sugerencias sobre mejoras de la iteración pasada, se recalca lo que se realizó de forma correcta y se realizan distintas actividades que sean dinámicas.