**GLOSARIO DE TERMINOS**

**Arquitectura de software**: es un conjunto de patrones que proporcionan un marco de referencia necesario para guiar la construcción de un software, permitiendo a los programadores, analistas y todo el conjunto de desarrolladores del software compartir una misma línea de trabajo y cubrir todos los objetivos y restricciones de la aplicación. Es considerada el nivel más alto en el diseño de la arquitectura de un sistema puesto que establecen la estructura, funcionamiento e interacción entre las partes del software.

**Actor**: no es más que un conjunto de roles que los usuarios de Casos de Uso desempeñan cuando interaccionan con estos Casos de Uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que hemos identificado los actores del sistema, tenemos identificado el entorno externo del sistema

**Casos de uso**: caso de uso representa la lista de tareas que pueden realizar los actores y está directamente relacionada con los requisitos del proceso empresarial. Los casos de uso son un reconocimiento de los requisitos que debe alcanzar el proyecto. Para documentar un caso de uso, defina los requisitos del objetivo, proporcione una presentación y liste los diferentes actores o roles para un caso de ejemplo dado.

**Reléase**; es una versión que se distribuye a los clientes.   
Cada reléase incluye nueva funcionalidad o está concebida para diferentes plataformas de hardware.

Artefacto:   es un producto tangible resultante del proceso de desarrollo de software.

**Roll-out**: “es lanzamiento” cuando se trata de comenzar a ofrecer en el mercado determinado producto o una versión determinada de un producto

**RSA**: es un sistema criptográfico de clave pública desarrollado en 1977. Es el primer y más utilizado algoritmo de este tipo y es válido tanto para cifrar como para firmar digitalmente.

**Sistema legacy**:  un sistema Legacy es aquel que ya no cumple con las necesidades del negocio, es aquel sistema que ya no puede recibir soporte y mantenimiento o simplemente un sistema que utiliza tecnología anticuada. Estos sistemas suelen ser difíciles de mantener, causan un alto costo para la empresa y pueden representar un riesgo de seguridad para el negocio, pero ¿por qué una empresa decidiría mantener sus sistemas legacy.

**SoapUI:** es una aplicación muy versátil que nos permite probar, simular y generar código de servicios web de forma ágil. En este tutorial introducimos esta herramienta, mostrando su instalación y un primer ejemplo de uso con un web service público.

**Testing**: como una investigación técnica de un producto bajo prueba con el fin de brindar información relativa a la calidad del software, a los diferentes actores involucrados en un proyecto. A partir de la información obtenida del testing se pueden tomar decisiones. Las decisiones pueden ser desde cuándo liberar un producto a producción, conociendo los riesgos que esto implica, hasta cómo mejorar las diferentes áreas dentro de la empresa. En definitiva, el testing es un agente de cambio, lo importante es interpretar la información obtenida para que todos los actores puedan actuar en forma oportuna donde sea necesario.

**UML:** por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el Object Management Group (OMG).