Lista de Chequeo: Arquitectura, Diseño y Modelado de Amenazas

- **Nivel 1 (Oportunista)**: En este nivel se conocen todos los componentes y las funciones para la cual se han concebido. Por ejemplo: componente de envío de correos, componente de autenticación, etc.

- **Nivel 2 (Estándar):** La arquitectura se establece completamente y el código es adecuado para esta. Por ejemplo: se define una arquitectura orientada a servicios y en el código se exponen ***API endpoints*** usando ***API REST***.

- **Nivel 3 (Avanzado):** Conocido el problema a solucionar, la arquitectura y el diseño propuestos se adaptan perfectamente para convertirse en solución.
  Por ejemplo: Aplicación en tiempo real, con una arquitectura de microservicios.

Descripción	Nivel		
	1	2	3
Verificar que todos los componentes de la aplicación se encuentran identificados y asegurarse que son necesarios.	X	X	X
Verificar que todos los componentes necesarios que no son parte de la aplicación (librerías de terceros, módulos externos, sistemas externos, etc) se han identificado.		X	X
Verificar que se ha definido una arquitectura de alto nivel para la aplicación.		X	X
Verificar que todos los componentes de la aplicación se definen de acuerdo a las funciones de negocio o de seguridad que proporcionan.			X
Verificar que todos los componentes que no son parte de la aplicación pero que son necesarios para su funcionamiento, sean definidos de acuerdo a las funciones de negocio o de seguridad que proporcionan.			X
Verificar que se ha realizado un modelo de amenazas para la aplicación en cuestión y que éste cubre riesgos asociados con la suplantación de identidad, manipulación, repudio, revelación de información y elevación de privilegios (***STRIDE***).			X
Verificar que todos los controles de seguridad (incluyendo las bibliotecas que llaman a servicios de seguridad externos) tienen una implementación centralizada.		X	X
Verificar que los componentes están separados unos de otros mediante controles de seguridad, tales como segmentación de la red, reglas de firewall, o grupos de seguridad basados en la nube.		X	X

Verificar que la aplicación tiene una clara separación entre la capa de datos, la capa de control y la capa de presentación, tal que las decisiones de seguridad pueden aplicarse en sistemas confiables.		X	X
Verificar que no hay ninguna lógica de negocio sensible, claves secretas u otra información propietaria en el código del lado del cliente.		X	X
Verificar que todos los componentes de la aplicación, bibliotecas, módulos, frameworks, plataformas y sistemas operativos se encuentran libres de vulnerabilidades conocidas		X	X