Lista de tablas

1. INTRODUCCION

Teniendo en cuenta los nuevos desafíos tecnológicos, la seguridad de la información y buenas prácticas que se deben aplicar en un proceso de desarrollo, adquisición y mantenimiento de software, se han convertido en elementos fundamentales para garantizar la integridad, confidencialidad y disponibilidad de la información. La creciente amenaza de ciberataques y brechas de seguridad ha hecho que el desarrollo seguro de software sea una prioridad ineludible para organizaciones o entidades de gobierno, y la UAESP a través de la Oficina TIC busca realizar la implementación de buenas prácticas.

A través de este documento se busca proporcionar a los desarrolladores, ingenieros de software, profesionales de seguridad, proveedores y concesionarios, una guía completa para comprender, implementar y mantener prácticas de desarrollo seguro en todas las etapas del ciclo de vida del software a fin de contribuir de manera significativa a la protección de datos sensibles y la confianza de los usuarios, al tiempo que reduce los riesgos y los costos asociados con posibles incidentes de seguridad.

2. OBJETIVO GENERAL

Definir los lineamientos para tener en cuenta durante el ciclo de vida de desarrollo seguro de software, permitiendo contar con aplicaciones con los mejores estándares de seguridad, calidad, usabilidad, accesibilidad y cumpliendo con la normatividad vigente.

3. ALCANCE

Aplica a todos los niveles de la Unidad Administrativa de Servicios Públicos que ejerzan labores o tengan responsabilidad en procesos de desarrollo, adquisición y mantenimiento de software.

4. SIGLAS Y DEFINICIONES

API: Application programming interface – (Interfaz de programación de aplicaciones).

DAST: Dynamic Application Security Testing – (Pruebas dinámicas de seguridad de aplicaciones).

DBA: Data base administrator – (Administrador de las bases de datos)

ISO: International Organization for Standardization – (Organización internacional de normalización).

MINTIC: Ministerio de las tecnologías de la información y las comunicaciones.

MSPI: Modelo de Seguridad y Privacidad de la Información.

NIST: National Institute of Standards and Technology – (Instituto Nacional de Estándares y Tecnología).

OTIC: Oficina de tecnologías de la información y las comunicaciones.

OWASP: Open web application Security Project – (Proyecto de seguridad de aplicaciones web).

SAMM: Software Assurance Maturity Model – (Modelo de madurez de aseguramiento de software).

SAST: Static application security testing – (Pruebas de seguridad de aplicaciones estáticas).

S-SDLC: Secure Software Development Live Cicle – (Ciclo de desarrollo de software seguro en vivo).

UAESP: Unidad Administrativa Especial de Servicios Públicos.

VCS: Version Control System – (Sistema de control de versiones)

Autenticación: Acto o proceso de confirmar que algo o alguien es quien dice ser.

Autorización: Proceso mediante el que el usuario obtiene los privilegios necesarios para poder acceder a los recursos o información.

Captcha: Completely Automated Public Turing test to tell Computers and Humans Apart (test de Turing completamente automático y público para distinguir ordenadores de humanos). Test utilizado por sitios y servicios web para comprobar si el usuario es un internauta humano y no un robot; consiste en identificaciones sencillas de letras, cifras o imágenes.

Confidencialidad: Es un principio fundamental de la seguridad de la información que garantiza el necesario nivel de secreto de la información y de su tratamiento, para prevenir su divulgación no autorizada cuando está almacenada o en tránsito.

Dast: Herramientas y técnicas para analizar la seguridad de aplicaciones de manera dinámica.

Desarrollo de software a la medida: Tipo de software desarrollado para cumplir con requerimientos específicos que no se encuentran cubiertos por software de propósito general.

Disponibilidad: Capacidad que garantiza que el software es operativo y accesible por los usuarios.

Fábrica de software: Empresa dedicada al desarrollo de software ya sea comercial o la medida.

Fiabilidad: Se define en términos estadísticos como la probabilidad de operación libre de fallos de un programa o software. Característica fundamental de los sistemas informáticos por la que se mide el tiempo de funcionamiento sin fallos.

Framework: Un entorno de trabajo, o marco de trabajo es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

GET: El método GET envía la información en la propia URL, estando limitada a 2000 caracteres. La información es visible por lo que con este método nunca se envía información sensible.

GIT: Sistema de control de versiones distribuido gratuito y de código abierto diseñado para manejar todo, desde proyectos pequeños hasta proyectos muy grandes con velocidad y eficiencia.

Hardware: Conjunto de componentes físicos de un sistema informático.

Hash: Son algoritmos que consiguen crear a partir de una entrada (ya sea un texto, una contraseña o un archivo, por ejemplo) una salida alfanumérica de longitud normalmente fija que representa un resumen de toda la información que se le ha dado

HTTPS: (HyperText Transfer Protocol Secure), Protocolo de transferencia de hipertexto) es un protocolo de comunicación de Internet que protege la integridad y la confidencialidad de los datos de los usuarios entre sus ordenadores y el sitio web.

Incidente: Cualquier evento que tenga el potencial de afectar la preservación de la confidencialidad, integridad, disponibilidad o valor de la información

Integridad: Capacidad que garantiza que el código del software, activos utilizados, configuraciones no puedan ser o no hayan sido modificados o alterados sin autorización.

Interoperabilidad: Capacidad de las organizaciones para intercambiar información y conocimiento en el marco de sus procesos para interactuar hacia objetivos mutuamente beneficiosos, con el propósito de facilitar la entrega de servicios digitales a ciudadanos, empresas y otras entidades, mediante el intercambio de datos entre sus sistemas.

Log: Registro de un evento ocurrido durante la operación de un sistema de información

Paiload: En informática y telecomunicaciones es el conjunto de datos transmitidos útiles, que se obtienen de excluir cabeceras, metadatos, información de control y otros datos que son enviados para facilitar la entrega del mensaje.

POST: Método que introduce los parámetros en la solicitud HTTP para el servidor. Por ello, no quedan visibles para el usuario. Además, la capacidad del método POST es ilimitada.

Principio de mínimo privilegio: El principio de mínimo privilegio consiste en reducir al mínimo el impacto de posibles fallos de seguridad reduciendo los permisos de las cuentas de usuario a los necesarios y limitando las cuentas de administrador a las estrictamente necesarias. Las cuentas de administrador son aquellas que tienen permisos para modificar los sistemas, actualizarlos, dar permisos a terceros o instalar nuevo software. Mediante un control de acceso, los empleados únicamente tendrán permiso para acceder a los servicios y gestionar aquella información que sea estrictamente necesaria para desarrollar su trabajo.

Resiliencia: Habilidad de un software para recuperarse y ajustarse a sí mismo en escenarios de estrés o interrupción, logrando ejecutar la tarea para la cuál ha sido diseñado.

Sanitizar: Proceso por el cual se controlan los parámetros que entran al sistema para asegurar el correcto funcionamiento de este.

Sast: Pruebas de seguridad de aplicaciones estáticas utilizadas para proteger el software mediante la revisión del código fuente del software para identificar las fuentes de vulnerabilidades.

Software: Programas y documentación de apoyo que permiten y facilitan el uso de la computadora además de automatizar procesos. El software controla el funcionamiento del hardware y el procesamiento de datos.

Software de propósito general: Dentro de esta categoría se agrupa el software desarrollado para el cumplimiento de un propósito puntual como, por ejemplo: ofimática, CRM, ERP, bases de datos entre otros.

SSL: (Secure Sockets Layer), capa de sockets seguros, es un protocolo para navegadores y servidores web que permite autenticar, cifrar y descifrar la información enviada a través de Internet.

Tester: Los probadores de software (también conocidos como testers, su denominación en inglés) planifican y llevan a cabo pruebas de software de los ordenadores para comprobar si funcionan correctamente. Identifican el riesgo de sufrir errores de un software, detectan errores y los comunican. Evalúan el funcionamiento general del software y sugieren formas de mejorarlo.

5. CICLO DE VIDA DEL DESARROLLO SEGURO DE SOFTWARE (S-SDLC)

Dentro del ciclo de vida de vida del desarrollo seguro se definen las siguientes fases:

- **Fase 1**: Planificación, análisis y diseño
- **Fase 2**: Desarrollo y pruebas
- **Fase 3**: Implementación, gestión de cambio y mantenimiento

5.1 FASE 1: PLANIFICACIÓN, ANÁLISIS Y DISEÑO

En esta fase se tienen en cuenta las tareas propias de la etapa de planificación definiendo el alcance, los requerimientos y el respectivo levantamiento de información tendiente a conocer en detalle el proceso en estudio. De igual manera se realiza el análisis de la información recolectada identificando los casos de uso, requerimientos funcionales y no funcionales de sistema. Por su parte en la etapa en la de diseño se debe definir de manera clara la infraestructura requerida para el proyecto y la disposición de los respectivos ambientes necesarios para el proyecto.

Durante esta fase se deben definir e identificar los objetivos y requisitos de seguridad, lo cual debería involucrar a los Oficiales de seguridad de la información y datos personales de la Entidad. Los requisitos de pueden definir teniendo en cuenta lo siguiente:

TABLA 1 REQUISITOS PARA TENER EN CUENTA EN FASE 1

ÍTEM	REQUISITO	DESCRIPCIÓN
1	Definición de base de datos	Se debe definir qué motor de base de datos se va a utilizar, su licenciamiento, restricciones y asegurar las conexiones entre el aplicativo y la base de datos.
2	Ambientes para el proyecto	Se deben definir la arquitectura y aprovisionar los ambientes de desarrollo, pruebas y producción de manera aislada tanto para aplicación como para base de datos de acuerdo con las necesidades.
		Los ambientes se deberán configurar con similares características a fin de prevenir comportamientos distintos en los diferentes ambientes.
		Se deberá garantizar la misma versión de motor de base de datos, librerías, complementos y demás utilitarios requeridos por la aplicación, en los diferentes ambientes.
		Las URL asignadas para los ambientes deben ser diferentes para cada uno, a fin de evitar confusión en las diferentes etapas del ciclo de vida de desarrollo.
3	Autenticación, roles y permisos	Se debe hacer uso del método de autenticación ya sea mediante contraseñas propias del software, directorio activo o cualquier otro modo implementado por la UAESP como mecanismo de seguridad para ingreso de los usuarios a los sistemas de información. En todos los casos

ÍTEM	REQUISITO	DESCRIPCIÓN
		se deben tener en cuenta las políticas de uso de contraseñas adoptadas en la Entidad.
		Definir la arquitectura de datos del software o sistema de información y su acceso, teniendo en cuenta el uso de privilegios, roles y perfiles de usuarios a fin de determinar permisos de escritura, lectura, modificación y eliminación.
4	Verificación, validación y protección de datos	Se debe asegurar el cumplimiento de la ley de datos personales y la normatividad vigente.
		Definir la clasificación de los datos que se manejarán en el software o sistema de información como confidenciales y públicos, tendiente a proteger los datos personales que se puedan almacenar en la base de datos.
5	Logs/auditoría	Se debe contemplar la implementación logs o auditoria, a fin de mantener registro de las acciones realizadas sobre el software y mantener trazabilidad de los eventos ocurridos en el sistema. Se debe mantener registro mínimo de la siguiente información: usuario que realiza la acción, evento o acción realizada, ubicación o dirección IP desde donde se realiza la conexión, fecha y hora de la ocurrencia del evento o acción.
6	Definir requisitos de la gestión de sesiones	Definir la gestión de sesiones garantizando que los usuarios autenticados puedan acceder a los recursos protegidos, mientras que los usuarios no autorizados tengan restringido dicho acceso.
		Se debe definir tiempos de inactividad del sistema a fin de cerrar sesiones de manera automática cuando se presente un periodo de inactividad o uso del software por parte del usuario.

ÍTEM	REQUISITO	DESCRIPCIÓN
7	Definir controles criptográficos	Se deben definir los controles criptográficos tanto para los datos almacenados como para las comunicaciones.
8	Interoperabilidad	En el caso de que el nuevo software genere información susceptible de compartir con otras Entidades, se debe asegurar la implementación del marco de interoperabilidad, a fin de desarrollar capacidades para el intercambio fácil, seguro y transparente de la información entre entidades públicas y de ser necesario, con entidades privadas, de acuerdo con lo establecido en el Marco de Interoperabilidad para Gobierno Digital de MinTIC y documentos concordantes.

FUENTE: UAESP 2023

Una vez reconocidos y aplicados los requisitos anteriores, en esta fase se debe aplicar la siguiente lista de chequeo tendiente a verificar las condiciones de desarrollo seguro de software.

TABLA 2 LISTA DE CHEQUEO FASE 1

LISTA DE CHEQUEO FASE I				
Nombre del Proyecto:			**Fecha**	
Responsable del Proyecto:				
CONFIDENCIALIDAD		**SI**	**NO**	**NA**
1	¿El aplicativo tiene páginas clasificadas privadas?			
2	¿El aplicativo requiere autenticación para todos los recursos y páginas excepto aquellas específicamente clasificadas como públicas?			
3	¿Se definieron cuáles de los datos que almacenará el aplicativo son privados y cuáles públicos?			

4	¿Se definieron los tipos de registro que debe generar el sistema?			
5	¿Se definieron los accesos a los recursos del sistema?			
6	¿Se definieron los perfiles de usuario con sus respectivos permisos?			
7	¿Se definió el modo de autenticación al ingreso del aplicativo?			
INTEGRIDAD		**SI**	**NO**	**NA**
8	¿Se valida todos los datos brindados por el usuario antes de procesarlos, incluyendo todos los parámetros, URLs y contenidos de cabeceras HTTP (por ejemplo, nombres de Cookies y valores)?			
9	¿Se contempló no difundir información sensible en respuestas de error, incluyendo detalles del sistema, identificadores de sesión o información de la cuenta?			
10	¿Se contempló controles para administración de sesiones?			
11	¿Se contempló que la función de logout debe terminar completamente con la sesión o conexión asociada y debe estar disponible en todas las páginas protegidas por autenticación?			
12	¿Se contempló el uso de registro de log's y que los datos del registro de log contengan información importante?			
13	¿Se contempló registrar en un log todas las fallas en los controles de acceso?			
14	¿Se contempló no guardar información sensible en logs, incluyendo detalles innecesarios del sistema?			
15	¿Se contempló asegurar que existan mecanismos para conducir un análisis de los logs?			
16	¿Se definieron las acciones a tomar si no se pueden registrar logs?			
17	¿Se contempló que el ambiente de desarrollo y pruebas sea configurado con la misma seguridad que el ambiente de producción?			

18	¿Se contempló no habilitar funcionalidades de completar automáticamente en aquellos formularios que contienen información sensible, incluyendo la autenticación?			
19	Se contempló la funcionalidad de captchas (verificar que el usuario que está accediendo a determinados datos es un humano y no una máquina)			
DISPONIBILIDAD		**SI**	**NO**	**NA**
20	¿Se contempló el lugar de almacenamiento de los logs?			
21	¿Se contemplaron los riesgos del proyecto en esta etapa?			
22	¿Se contemplaron acciones para minimizar los riesgos del proyecto?			
23	**Observaciones:**			

FUENTE: Ajuste propio basado en la definida por el Ministerio de salud y protección social.

5.1.1 Control de versiones

Para asegurar la adecuada gestión del proyecto en relación con el control de versiones, el equipo de desarrollo debe hacer uso de la herramienta GIT establecida por la Oficina TIC de la UAESP de acuerdo con el procedimiento GTI-PC-18 Gestión de arquitectura de tecnologías de la información, actualizando el código y la documentación relacionada conforme evoluciona el proyecto según lo establecen las buenas prácticas. La principal ventaja de este sistema es que permite rastrear y gestionar los cambios realizados en un archivo o grupo de archivos, permitiendo el análisis de los cambios y de ser necesario revertirlos sin ocasionar errores.

5.2 FASE 2: DESARROLLO Y PRUEBAS

En esta fase se deben tener en cuenta las buenas prácticas de codificación y requisitos de seguridad. De igual manera se deben atender acciones propias para la aplicación

del plan de pruebas funcionales y de aceptación tendiente a la aprobación por parte del área funcional o solicitante.

5.2.1 Desarrollo

Al iniciar la etapa de desarrollo del sistema de información, se deben tener en cuenta los siguientes requisitos:

TABLA 3 REQUISITOS PARA TENER EN CUENTA EN FASE 2, ETAPA DE DESARROLLO

ÍTEM	REQUISITO	DESCRIPCIÓN
1	Ambientes para el proyecto	Se debe aprovisionar los ambientes de desarrollo, pruebas y producción de manera aislada tanto para aplicación como para base de datos de acuerdo con las necesidades.
		Los ambientes deben estar configurados con similares características a fin de prevenir comportamientos distintos en los diferentes ambientes.
		Se debe verificar que se cuenta con la misma versión de motor de base de datos, librerías, complementos y demás utilitarios requeridos por la aplicación, en los diferentes ambientes.
		Las URL asignadas para los ambientes deben ser diferentes para cada uno, a fin de evitar confusión en las diferentes etapas del ciclo de vida de desarrollo.
		Los desarrolladores deben realizar su trabajo exclusivamente en el ambiente de desarrollo aprovisionado por la UAESP. No deben realizar tarea alguna en otros ambientes o no autorizados por la Entidad.

ÍTEM	REQUISITO	DESCRIPCIÓN
2	Autenticación y gestión de sesiones	Hacer uso de los mecanismos de autenticación definidos en la fase 1, siempre atendiendo la política de contraseñas adoptado por la Entidad.
		Permitir acceso a los usuarios atendiendo el diseño de roles y permisos establecidos en la fase 1, con el fin de garantizar que usuarios no autorizados tengan acceso a información no autorizada.
		Verificar que las sesiones se invaliden cuando el usuario cierra la sesión.
		Verificar que las sesiones se invaliden luego de un período determinado de inactividad.
		Todas las páginas dentro del software o aplicación deben tener restricción de acceso validando la sesión actual y los respectivos permisos de usuario.
3	Validación datos de entrada y salida	Se debe validar la entrada de cada campo teniendo en cuenta el tipo de dato y su longitud máxima definida.
		Los datos que provienen de parámetros o valores fijos, preferiblemente se debe mostrar en la interfaz del software o aplicación con listas desplegables o cualquier otro objeto en el cual le permita al usuario final seleccionar los valores, a fin de reducir posibilidad de errores de entrada al digitar datos.
		Se debe garantizar que la visualización de los datos de salida, se presenten de forma correcta teniendo en cuenta la longitud, formato y demás características que faciliten la lectura de la información.

ÍTEM	REQUISITO	DESCRIPCIÓN
4	Protección de datos personales	No utilizar campos ocultos para el almacenamiento de datos sensibles o confidenciales, eventualmente se podría exponer la información.
		En caso de requerirse uso de datos personales, se debe contar con mecanismo a través del software o aplicación para la respectiva autorización de uso.
		En caso de requerir el uso de datos confidenciales o sensibles, se deben utilizar mecanismos de cifrado o protección de información ya sea para el almacenamiento o visualización a través de la interfaz del software o aplicación.
5	Protección de información, URL y datos transmitidos	Verificar que en las URL no se revelen datos que permitan vulnerar la aplicación como datos de sesiones, contraseñas, mensajes de error con identificadores que generen brechas de seguridad del sistema.
		Se debe garantizar que los datos personales y demás información que comprometa la seguridad del sistema, que sean enviados como parámetros, deben estar debidamente protegidos de tal manera que no expongan a través de URL, variables de entorno, archivos o cualquier otro medio de intercambio.
		No se debe revelar al usuario final información propia del sistema como nombre de servidores, bases de datos, contraseñas, nombre de objetos y demás recursos que expongan la seguridad.
6	Actualización del sistema de	Se debe actualizar permanentemente en la herramienta de control de versiones disponible en la UAESP, el código fuente y documentación con el fin de mantener historial y

ÍTEM	REQUISITO	DESCRIPCIÓN
	control de versiones	control del versionamiento durante la etapa de desarrollo del proyecto.

FUENTE: UAESP 2023

5.2.2 Buenas prácticas en etapa de desarrollo

A continuación, se enumeran las recomendaciones que se deben tener en cuenta en la fase de desarrollo:

- Haga uso de nombres descriptivos, para la declaración de variables, en otras palabras, que hagan referencia a su nombre y no a su tipo.
- En lo posible no usar variables globales, debido a que atacantes podrían acceder a estas y modificar su valor.
- Las variables siempre deben ser inicializadas.
- Se debe verificar que los logs almacenen información relevante sobre las tablas y transacciones críticas, permitiendo almacenar y consultar información como: usuario, fecha, hora, MAC de la máquina, dirección IP, nombre del host, acción ejecutada (ejecución, borrado, creación, modificación), tabla modificada etc.
- Se debe comentar el código de manera que describa las funcionalidades que se están programando.
- Se recomienda controlar en número de instrucciones anidadas.
- Se recomienda la reutilización de código fuente.
- Se recomienda no mezclar datos con código, para evitar errores en el procesamiento y conllevar a fallas en la seguridad.

- Evitar el uso de métodos con muchos parámetros, se debe considerar la creación de una clase que tenga las propiedades requeridas.

- Se recomienda no hacer comparaciones explícitas con expresiones booleanas (falso y verdadero), se sugiere asignar la condición a una variable y utilizar esta variable en las comparaciones, adicional se sugiere nombrar las variables de forma afirmativa.

- Se deben validar todos los parámetros de las (API) interfaces de programación de aplicaciones exportadas, corroborando que sean válidos.

- Se recomienda el uso de API criptográficas de firmas reconocidas (Microsoft, IBM, etc.)

- Se recomienda el uso de métricas sobre seguridad en las aplicaciones.

- Si una funcionalidad se debe implementar en diferentes aplicaciones, es recomendable crear un script, rutina, servicio o función que pueda ser reutilizable en cualquier aplicación

5.2.3 Pruebas

A continuación, se describen los requisitos que se deben tener en cuenta durante la etapa de pruebas del sistema:

TABLA 4 REQUISITOS PARA TENER EN CUENTA EN FASE 2, ETAPA DE PRUEBAS

ÍTEM	REQUISITO	DESCRIPCIÓN
1	Pruebas funcionales	Definir plan de pruebas de acuerdo con los requisitos funcionales aprobados.
		En caso de requerir uso de datos personales en la ejecución de las pruebas, se deberá contar con permiso de utilización definiendo tiempo de vigencia de la autorización.

ÍTEM	REQUISITO	DESCRIPCIÓN
		En lo posible se deben utilizar datos dummies o ficticios para la realización de las pruebas sin perjuicio de la calidad de la prueba.
		Documentar la ejecución de las pruebas de acuerdo con el formato establecido por la Oficina TIC, indicando los criterios de aceptación y si ésta resulta satisfactoria o no. En caso de resultar *No satisfactoria* la prueba, se realizará en el software las correcciones pertinentes y se repetirá la respectiva prueba hasta cumplir los criterios de aceptación planteados. Esto tendiente a descartar errores que se presenten al hacer uso del software.
		En caso de tratarse de un mantenimiento del software, las pruebas deben encaminarse a verificar todas las funcionalidades del sistema que se puedan ver afectadas por la nueva funcionalidad.
		Validar el ingreso de datos en los campos verificando el tipo de dato, longitud, uso de caracteres especiales y demás características propias de los valores de cada campo, de tal manera que corresponda con lo requerido en el software y se mantenga la integridad referencial de la información.
2	Pruebas de seguridad	Se debe validar que el sistema de autenticación sea el definido para la UAESP y se cumplan las políticas de seguridad de contraseñas.
		Se deben verificar los niveles de acceso para constatar que los usuarios autorizados puedan acceder únicamente a los módulos o funcionalidades de acuerdo con su rol.
		Verificar la vigencia de las sesiones y bloqueos por tiempos de inactividad.

ÍTEM	REQUISITO	DESCRIPCIÓN
		Validar que los mensajes de error, URL, archivos y demás medios de transferencia de datos en la aplicación, no revele información que comprometa la seguridad del sistema.
		El registro de logs debe contener información útil al momento de una auditoría, atendiendo las políticas de uso de datos sensibles en caso de ser necesario.
		Si se utilizan complementos, librerías, utilidades o herramienta de terceros en el software, se debe garantizar la última versión estable ofrecida por el fabricante y analizar las posibles vulnerabilidades a las que se pueda exponer el software desarrollado, derivado del uso de estas herramientas.
		Siempre que sea posible, se debe realizar análisis de código estático, priorizando la corrección de vulnerabilidades o errores (bugs) críticos.
3	Pruebas de carga	Se deben realizar pruebas de carga que permitan evaluar cual es el número máximo de peticiones por unidad de tiempo que le es posible atender al aplicativo y con base en esta información realizar los ajustes necesarios.

FUENTE: UAESP 2023

Derivado del resultado de las pruebas, una vez sean satisfactorias, se gestionará con el usuario funcional la aceptación del producto y se concertará la salida a producción. Una vez reconocidos y aplicados los requisitos para las etapas de desarrollo y pruebas, así como las buenas prácticas de desarrollo, se debe aplicar la siguiente lista de chequeo:

TABLA 5 LISTA DE CHEQUEO FASE 2

LISTA DE CHEQUEO FASE II

Nombre del Proyecto:		Fecha		
Responsable del Proyecto:				
CONFIDENCIALIDAD		**SI**	**NO**	**NA**
1	¿Se implementaron las páginas que fueron definidas como clasificadas privadas?			
2	¿Se implementó la autenticación para todos los recursos y páginas definidas excepto aquellas que específicamente fueron clasificadas como públicas?			
3	¿Los datos almacenados definidos como privados cuentan con las restricciones correspondientes?			
4	¿Se implementaron los tipos de registro que debe generar el sistema?			
5	¿Fueron implementados los accesos a los recursos del sistema, con base en los requerimientos?			
6	¿Se implementaron los perfiles de usuario con sus respectivos permisos, con base en los requerimientos?			
7	¿Se implementó el modo de autenticación definido en la fase 1?			
INTEGRIDAD		**SI**	**NO**	**NA**
8	¿Se verificó que el ambiente de desarrollo y pruebas esté configurado con la misma seguridad que el ambiente de producción?			
9	¿Se validaron todos los datos brindados por el usuario antes de incluirlos en el aplicativo?			
10	¿Se realizaron pruebas para validar que los mensajes de error no difundan información sensible ni detalles del sistema, identificadores de sesión o información de la cuenta?			
11	¿Se implementó el control de sesiones?			

12	¿Se realizaron pruebas para validar la función de logout que termine completamente con la sesión o conexión asociada y estar disponible en todas las páginas definidas que deberían ser protegidas por autenticación?			
13	¿Se implementó los eventos definidos para el log para el registro de eventos o sucesos del sistema?			
14	¿En las pruebas se verificó que el registro del log tenga todas las fallas en los controles de acceso?			
15	¿En las pruebas se verificó que no se guarde información sensible, ni detalles innecesarios del sistema?			
16	¿Las pruebas realizadas verificaron que no exista la funcionalidad de completar automáticamente en los formularios que contienen información sensible, incluyendo la autenticación?			
17	¿Se validó la funcionalidad captchas?			
18	¿Se cumple con la política de contraseñas definida?			
19	¿Los desarrolladores sólo realizaron su trabajo en el ambiente de desarrollo, nunca en otros ambientes directamente?			
20	¿Los nombres de dominio o URL para los ambientes de producción, pruebas y desarrollo, son diferentes?			
21	¿Para las pruebas se utilizó datos que no eran reales, anonimizados o se obtuvo autorización para la utilización de datos personales en etapa pruebas?			
22	¿En general las pruebas realizadas contemplaron los lineamientos de seguridad estipulados en el presente documento?			
DISPONIBILIDAD		**SI**	**NO**	**N.A**
23	¿Se contó con el ambiente de desarrollo?			
24	¿Se contó con el ambiente de pruebas?			

25	¿La seguridad de los ambientes de desarrollo y pruebas tenían el mismo nivel de seguridad, complementos, librerías y demás herramientas necesarias que el de producción?			
26	¿Se verificó el almacenamiento de los logs?			
27	¿Se solicitaron los permisos para el acceso a los ambientes de acuerdo con el perfil de cada persona?			
28	¿Fueron asignados los permisos de acceso a los ambientes de acuerdo con lo solicitado?			
29	**Observaciones:**			

Fuente: Ajuste propio basado en la definida por el Ministerio de salud y protección social

5.3 FASE 3: IMPLEMENTACIÓN, GESTIÓN DE CAMBIO Y MANTENIMIENTO

Una vez finalizadas la etapa de pruebas y se obtenga la aceptación por parte del área solicitante o usuario funcional, se dará inició a la puesta en producción del sistema de información, el proceso de gestión de cambio y analizar el mantenimiento del software o atención de incidencias relacionadas con el sistema de información.

En esta fase de deben tener en cuenta los siguientes requisitos.

TABLA 6 REQUISITOS PARA TENER EN CUENTA EN FASE 3

ÍTEM	REQUISITO	DESCRIPCIÓN
1	Implementación	Realizar despliegue a ambiente de producción previa revisión de las condiciones de seguridad tanto en servidores de aplicación como de base de datos.
		Realizar configuración de la aplicación, servidor de base de datos, complementos, librerías y demás herramientas requeridas por el software para su correcta operación,

ÍTEM	REQUISITO	DESCRIPCIÓN
		garantizando el cumplimiento de las políticas de seguridad adoptadas en la Entidad.
		Realizar pruebas de operación en el ambiente de producción antes de ser entregado formalmente al área solicitante.
		Implementar la política de backup de la base de datos del nuevo sistema de información, de acuerdo con el procedimiento establecido por la UAESP.
2	Gestión de cambio	Realizar jornada de uso y apropiación del nuevo sistema de información o funcionalidad desarrollada en el caso de mantenimiento de software existente.
		Realizar entrega de la respectiva documentación del proyecto de desarrollo como manual de arquitectura de la solución (Arquitectura, diagramas de casos de uso, modelo de datos, diccionario de datos, diagramas de red y los demás que se consideren necesarios), Manual técnico, manual de usuario y demás información que se considere pertinente para documentar de manera precisa el sistema de información.
3	Actualización del sistema de control de versiones	Se debe verificar el adecuado almacenamiento de las diferentes versiones del software y su funcionamiento.

FUENTE: UAESP 2023

Una vez se atiendan los requisitos de la fase 3, se debe aplicar la siguiente lista de chequeo:

TABLA 7 LISTA DE CHEQUEO FASE 3

LISTA DE CHEQUEO FASE III				
Nombre del Proyecto:		**Fecha**		
Responsable del Proyecto:				
INTEGRIDAD		**SI**	**NO**	**NA**
1	¿Se cumplió con las listas de comprobación de la fase II en su totalidad?			
2	¿La memoria se libera de manera adecuada a la salida de una función y de todos los puntos de finalización?			
3	¿Los mantenimientos o resolución de incidencias contemplaron los lineamientos de seguridad?			
CONFIDENCIALIDAD		**SI**	**NO**	**NA**
4	¿La sensibilización del aplicativo o desarrollo abarcó los temas de seguridad como: política de seguridad de la información, confidencialidad y uso adecuado?			
5	¿La sensibilización se realizó a la totalidad de las personas o grupo significativo que van a utilizar el sistema?			
6	¿Se compartió el manual de usuario y está disponible para su consulta?			
DISPONIBILIDAD		**SI**	**NO**	**NA**
7	¿Los aplicativos contemplan mecanismos de seguridad relacionados con prevención?			
8	¿Los aplicativos contemplan mecanismos de seguridad relacionados con recuperación?			
9	¿Los aplicativos contemplan mecanismos de seguridad relacionados con detección?			
10	**Observacione**s:			

FUENTE: Ajuste propio basado en la definida por el Ministerio de salud y protección social

6. ACOMPAÑAMIENTO TÉCNICO EN LA ADQUISICIÓN DE SOFTWARE

El proceso de adquisición de software de la Entidad debe atender a las necesidades y obtener el mayor valor por el dinero público, por esta razón la entidad debe entender claramente y conocer:

- Cuál es su necesidad y cómo puede satisfacerla
- Cómo y quienes pueden proveer los bienes y servicios que necesita
- Contexto en el cual los posibles proveedores desarrollan su actividad

El Decreto 1082 de 2015 en su artículo 2.2.1.1.1.6.1. específica “La Entidad Estatal debe hacer, durante la etapa de planeación, el análisis necesario para conocer el sector relativo al objeto del Proceso de Contratación desde la perspectiva legal, comercial, financiera, organizacional, técnica, y de análisis de Riesgo. La Entidad Estatal debe dejar constancia de este análisis en los Documentos del Proceso”. En total sintonía con este Decreto la oficina TIC acompaña a las diferentes dependencias de la Entidad, prestando asesoría técnica y análisis de riesgos de seguridad de la información.

A continuación, se presentan las recomendaciones que a la oficina TIC le corresponden asesorar:

5.4 ACOMPAÑAMIENTO TÉCNICO

La oficina TIC activamente asesora y acompaña durante la fase de estudios previos y análisis de mercado al solicitante para apoyarlo de manera técnica sobre los requisitos que debe cumplir el software a adquirir, teniendo en cuenta puntos como: compatibilidad con la infraestructura presente en la Entidad, interfaces con otros sistemas, seguridad y privacidad de la información, disponibilidad de la infraestructura necesaria para su operación, protección de datos personales y demás requerimientos técnicos aplicables.

En este proceso de acompañamiento técnico de la oficina TIC contempla las siguientes actividades:

- Asesorar a la dependencia en revisar si en el mercado existen aplicaciones ya desarrolladas que atiendan las necesidades planteadas, en el entendido que estas por estar ya desarrolladas son más económicas, pueden tener niveles de madurez y seguridad mayores y su despliegue en tiempo es menor.

- Si la dependencia plantea la instalación de software libre se debe realizar un análisis de sus capacidades de funcionalidad, interoperabilidad, seguridad de la información, entre otros.

- Se debe asesorar a la dependencia encargada de la contratación sobre determinar la complejidad del proyecto (baja, media o alta) teniendo en cuenta dimensiones como:

  - ✓ Urgencia e importancia del proyecto
  - ✓ Duración del proyecto
  - ✓ Costos del proyecto
  - ✓ Equipo asociado al proyecto
  - ✓ Composición del equipo de proyecto
  - ✓ Alcance del proyecto
  - ✓ Importancia estratégica del proyecto
  - ✓ Nivel de impacto dentro de la organización
  - ✓ Riesgos asociados al proyecto
  - ✓ Limitaciones y dependencias del proyecto

Sobre los perfiles de quienes van a realizar el proceso de despliegue o instalación del software, se debe tener en cuenta los requisitos habilitantes definidos por la legislación vigente.

La entidad debe tener en cuenta las siguientes recomendaciones de manera general para la adecuada estructuración del proyecto:

- Que el objetivo del proyecto de desarrollo de software está alineado con los objetivos de mayor nivel de la organización.
- El software desarrollado cumple con las expectativas de la entidad en la solución de un problema, cubrimiento de una necesidad o en la explotación de la oportunidad.
- La entidad percibe el valor del software desarrollado desde etapas tempranas del inicio del proyecto.
- El software cumple con los requerimientos estipulados en términos de calidad, usabilidad, funcionalidad, interoperabilidad y seguridad.
- El proyecto se desarrolla dentro de los márgenes de tiempo y recursos presupuestados.
- En el caso de requerir soluciones de software a través de la modalidad de concesión, y una vez esta finalice, el concesionario deberá entregar totalmente funcional en la infraestructura tecnológica de la UAESP, la última versión estable de la aplicación, así como la totalidad de los datos que se hayan producido durante su periodo de operación para efectos de consulta y visualización a perpetuidad.

Es importante atender las recomendaciones establecidas por Colombia Compra Eficiente, MinTic y las buenas prácticas asociadas.

7. USABILIDAD Y ACCESIBILIDAD

Para el desarrollo o adquisición de software de ambiente web, se debe involucrar al líder de la política de Gobierno Digital y se deben tener en cuenta consideraciones de usabilidad y funcionalidades de accesibilidad que se indican en la Política de Gobierno Digital de MinTIC y los anexos de la resolución 1519 de 2020.

7.1 USABILIDAD

La usabilidad es un atributo relacionado con la facilidad de uso. Se refiere a la facilidad con la que las personas pueden utilizar un software a través de su interfaz. Esto va de la mano con un diseño intuitivo, eficiente, accesible del sistema. Para evaluar la usabilidad del software se debe utilizar la siguiente lista de chequeo

TABLA 8 LISTA DE CHEQUEO USABILIDAD

LISTA DE CHEQUEO USABILIDAD				
Nombre del Proyecto:		**Fecha**		
Responsable del Proyecto:				
USABILIDAD		**SI**	**NO**	**NA**
1	¿No se evidencian errores de ortografía o redacción en el sistema de información?			
2	¿La secuencia o pasos lógicos para el uso del software es amigable?			
3	¿El usuario reconoce con facilidad la ruta en la cual está ubicado dentro del software?			
4	¿El diseño gráfico del sistema se conserva en todos los sitios de navegación?			
5	¿Se cuenta con un diseño ordenado y limpio?			
6	¿Los enlaces, botones, menús, etiquetas, entre otros, indican de manera clara el contenido al cual conducen?			

7	¿No se evidencian opciones o enlaces rotos?			
8	¿La alineación de los textos están alineados a la izquierda?			
9	¿La longitud de los textos no supera los 100 caracteres?			
10	¿El contenido se puede observar en la pantalla de manera correcta o se tienen objetos de desplazamiento vertical y horizontal para la visualización completa?			
11	¿La visualización en diferentes dispositivos se hace de manera correcta, es decir, se ajusta el contenido a la pantalla de visualización?			
12	¿Al momento de diligenciar información, el usuario puede identificar los campos que son de obligatorio diligenciamiento?			
13	¿El nombre del campo corresponde al espacio dispuesto para el diligenciamiento de la información?			
14	¿El software realiza la tarea esperada o cumple los requisitos funcionales definidos?			
15	¿Se proporcionan ayudas o documentación clara que ayuda al usuario final a hacer uso del software?			
16	¿Los mensajes de error están redactados de manera que sea entendible por el usuario final?			

FUENTE: UAESP 2023

7.2 ACCESIBILIDAD

Son las condiciones y características de los contenidos dispuestos en medios digitales para que puedan ser utilizados por la mayoría de los ciudadanos independientemente de sus condiciones tecnológicas o del ambiente, e incluyendo a las personas con discapacidad.

Se debe procurar el cumplimiento de la normatividad vigente en materia de accesibilidad para el software desarrollado o adquirido por la Entidad, en especial lo dispuesto en la resolución 1519 de 2020 emanada por el Ministerio de las Tecnologías de la Información y las Comunicaciones. Lo anterior teniendo en cuenta la finalidad del software, arquitectura y demás características propias que lo hagan adecuado para implementar pautas accesibles.

8. REFERENCIAS BIBLIOGRÁFICAS

- Ministerio de las Tecnologías de la Información y las Comunicaciones (Mayo 20023). Guía general. MAE.G.ASI - DOMINIO DE ARQUITECTURA SISTEMAS DE INFORMACIÓN. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.mintic.gov.co/arquitecturaempresarial/630/articles-237650_recurso_1.pdf

- Ministerio de las Tecnologías de la Información y las Comunicaciones (2023). Arquitectura empresarial Colombia. Gobierno Digital. https://mintic.gov.co/arquitecturaempresarial/portal/

- Ministerio de salud y protección social (2022). Lineamientos de Desarrollo Seguro de Software. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.minsalud.gov.co/Ministerio/Institucional/Procesos%20y%20procedimientos/CVSS01.pdf

- Ministerio de las Tecnologías de la Información y las Comunicaciones (2018). Kit Guía de usabilidad.

- Ministerio de las Tecnologías de la Información y las Comunicaciones (2019). Marco de interoperabilidad para Gobierno Digital.

9. CONTROL DE CAMBIOS

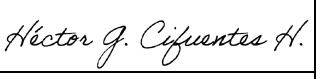
TABLA 9 CONTROL DE CAMBIOS

Versión	Fecha	Descripción de la modificación
01	18/11/2021	Creación del documento para establecer los lineamientos del ciclo de desarrollo de software.
02	12/01/2024	Actualización del documento en cuanto a los recursos de la Entidad, las mejores prácticas de desarrollo y se incluyen lineamientos de usabilidad y accesibilidad.

FUENTE: UAESP 2023

10. AUTORIZACIONES

TABLA 10 AUTORIZACIONES

	NOMBRE	CARGO	FIRMA
Elaboró	Héctor Gonzalo Cifuentes Hernández	Profesional Especializado – Oficina TIC	
	Sayra Paola Nova Murcia	Profesional especializado (E) – Oficina TIC	
	Juan Sebastian Perdomo Méndez	Profesional universitario – Oficina TIC	
Revisó	Cesar Mauricio Beltrán López	Jefe Oficina de Tecnologías de Información y las Comunicaciones	
	Luz Mary Palacios Castillo	Profesional Oficina Asesora de Planeación	
Aprobó	Yesly Alexandra Roa Mendoza	Jefe Oficina Asesora de Planeación	Alexandra Roa Mendoza Firmado digitalmente por Alexandra Roa Mendoza Fecha: 2024.01.12 09:03:48 -05'00'