Evaluación 4: Metodologías de Desarrollo Seguro Ponderación (30%)

**Unidad de Aprendizaje 4:**

Evalúa el uso de metodologías de desarrollo seguro, para proteger la integridad de la información, considerando SAMM y SDL.

**Aprendizaje esperado**

4.1 Evalúa librerías externas de seguridad en Python, para proteger la integridad, confiabilidad y disponibilidad de la información, según el estándar de la industria.

**Actividades**

1. Desarrollar Algoritmo.
2. Aplicar estructuras de datos para almacenar

(listas, tuplas, diccionarios)

1. Aplicar contadores.
2. Aplicar acumuladores.
3. Aplicar estructuras de control de repetición.
4. Aplicar estructuras de decisión.
5. Aplicar validaciones.
6. Aplicar Funciones.
7. Controlar excepciones.
8. Presentar resultados.

**Evaluación**

* La evaluación tiene una ponderación del 30%.
* Evaluación 4 será evaluada con rúbrica.
* El plazo de desarrollo y envío es de 1 semana.

|  |  |  |  |
| --- | --- | --- | --- |
| **INTEGRANTE(S)** | 1.  2. | | |
| **SECCION** |  | **FECHA** |  |
| **PUNTOS** |  | **NOTA** |  |

1. **Presentación**

A continuación, te invitamos a analizar un caso propuesto por el docente, con la finalidad de genera la propuesta de solución a través del desarrollo de funciones con Python, implementando, además, SAMM y SDL.

El objetivo de la evaluación es aplicar los contenidos aprendidos sobre el lenguaje de programación Python en las unidades de la asignatura y además complementando su trabajo con buenas prácticas de desarrollo adquiridas.

1. Criterios de evaluación

Las actividades consideran los siguientes criterios de evaluación:

4.1.1 Explica las metodologías de desarrollo seguro SAMM, de acuerdo con la integridad, confidencialidad y disponibilidad.

4.1.2 Explica metodología de desarrollo seguro SDL, de acuerdo con de acuerdo con la integridad, confidencialidad y disponibilidad.

4.1.3 Distingue buenas prácticas de desarrollo seguro, considerando la sanitización de variables, estructuras de datos, anonimización y seudonimización.

4.1.4 Evalúa script de desarrollo seguro, considerando las metodologías SDL, SAMM y las librerías Python de repositorios oficiales.

1. **Instrucciones**

* **Las copias serán evaluadas con nota 1,0. Estas copias incluyen archivos iguales entre compañeros o copias de alguna Inteligencia Artificial.**
* La evaluación tiene un plazo de desarrollo y envío de 1 semana (fecha y horario indicado por docente)
* La evaluación tiene 100 puntos en total.
* La nota 4.0 se obtiene logrando un 60% del puntaje total (60 puntos).
* Se exige por parte de los alumnos(as) una conducta ética y responsable.
* **Debe subir el desarrollo de la actividad en la tarea disponible para ello en el ambiente aprendizaje: "Evaluación 4".**

1. **Actividades**
2. Conformar grupos de trabajo con un máximo de **2 estudiantes**, los cuales pueden ser organizados por ustedes. Pero en caso de que no logren organizarse el docente tiene la facultad de crear los grupos de trabajo.
3. Una vez conformados los grupos, el docente asignará la temática a cada grupo. Después de eso deberán analizar los requerimientos en conjunto para poder crear un bosquejo de lo que será el programa asociado al tema.
4. El desarrollo consiste en la creación de 2 productos entregables, los cuales se detallan a continuación.

**PRODUCTO 1:** desarrollo del código en Python asociado a la temática asignada por docente en donde debe aplicar los contenidos revisado en clases sobre validación de datos, condiciones, excepciones, funciones, estructuras de almacenamiento de datos (arreglos), y lo relacionado con el lenguaje de programación pertinente para crear el programa (*indicaciones más abajo).*

**PRODUCTO 2:** informe sobre la implementación de las metodologías SAM, SDL y buenas prácticas de programación aplicadas en su programa codificado (*indicaciones más abajo).*

1. **Consideraciones de los productos solicitados**

**CONSIDERACIONES PRODUCTO 1 (CODIGO EN PYTHON)**

1. Observe la tabla para poder conocer el detalle de la temática que le fue asignada, en ella podrá conocer además los datos obligatorios que debe solicitar para completar el desarrollo de su programa. Se sugiere que en la misma clase donde conozca estas indicaciones pueda consultar al docente por las especificaciones, dudas o consultas sobre la temática o datos para poder así aplicar de forma correcta en el código.



1. Conocida la temática y los datos para pedir en el programa podrá comenzar a desarrollar su código con el lenguaje de programación Python en el software Visual Studio Code, teniendo en cuenta las especificaciones indicadas a continuación:
   1. El menú inicial se debe desplegar al momento de ejecutar el programa:

1. Iniciar Sesión (vea el ítem 2.2)

2. Salir (debe permitir terminar el programa)

* 1. El menú de usuario se desplegará dependiendo del usuario con el cual se inicie la sesión:

El usuario podrá ingresar con 3 usuarios diferentes *(ver ítem 4 para el detalle)*, para el caso que el usuario no este registrado (que no sea uno de los 3 disponibles) no podrá ingresar (se deja un mensaje).

Para los 3 usuarios con los cuales podría ingresar, considere que cada uno tendrá accesos distintos:

* Si ingresa con usuario **ADMIN** deberá acceder a un menú de opciones:

(1. Registrar. 2. Listar, 3. Estadística, 4. Cerrar Sesión).

* Si ingresa con usuario **OPERADOR** deberá acceder a un menú de opciones:

(1. Listar, 2. Cerrar Sesión).

* Si ingresa con usuario **GESTOR** deberá acceder a un menú con opciones:

(1. Estadística, 2. Cerrar Sesión).

1. Para el desarrollo del código debe generar las funciones necesarias para realizar el desarrollo de la actividad, según lo practicado en clases:

* iniciar\_sesion()
* cerrar\_sesion()
* registrar\_datos()
* listar\_datos()
* estadistica()
* salir()
* (entre otras más que considere necesarias).

1. Para iniciar sesión debe ingresar el nombre de usuario y la contraseña, considerando los siguientes perfiles:

* admin / inacap ("admin" es el nombre de usuario, mientras que "inacap" es la contraseña).
* operador / 123 ("operador" es el nombre de usuario, mientras que "123" es la contraseña).
* gestor / 321 ("gestor" es el nombre de usuario, mientras que "321" es la contraseña).

1. Cuando la opción del menú sea incorrecta, muestre el mensaje "Opción Incorrecta en Menu XXXX", donde la XXXX debe reemplazarla por el nombre del menú en el cual se encuentra.
2. De no haber registros en las estructuras de datos (listas, tuplas y/o diccionarios) cuando se digiten las opciones Listar y Estadística, muestre el mensaje de error correspondiente "No existen registros en el almacenamiento".
3. La opción 3. Estadística debe presentar los cálculos:

* Cantidad de registros según el dato 4.
* Suma de valores $(dato 6) evaluando como condición las opciones del dato 4.

1. La opción 4. Salir debe presentar el mensaje "Programa Finalizado!!" y luego detener la ejecución, pero solo después de confirmar la salida.
2. Al listar los registros, NO muestre los números de datos que son elegidos, sino más bien, lo que representan (textos descriptivos).
3. Al término de cada actividad debe volver al menú en el cual se encuentra, excepto cuando se quiera cerrar la sesión y/o salir del programa.
4. Al ir ingresando los datos para un nuevo registro en el programa, indique cuál es el número de registro (índice) según los ingresos que ya se hayan creado.
5. Implemente uso de while y try/except para validar el ingreso de datos según la temática que le fue asignada.

**CONSIDERACIONES PRODUCTO 2 (INFORME)**

1. El segundo producto de esta evaluación corresponde a un informe sobre las medidas de seguridad de datos que aplicó en el desarrollo de su código Python del programa. Este informe se sugiere desarrollar posterior a la creación del código, ya que, se debe analizar lo desarrollado.
2. El informe debe presentarse en formato digital.
3. El informe debe presentarse en formato Microsoft Word.
4. El informe deberá considerar los siguientes ítems:

* **Portada Institucional** (Nombre de integrantes, Nombre del tema asignado, Nombre de asignatura, Código de sección, Fecha de entrega).
* **Índice** (con los contenidos, numero de hoja y títulos principales de su desarrollo)
* **Introducción** (mínimo 150 palabras para presentar y explicar en qué consiste los principales temas).
* **Objetivos** (Los objetivos propuestos deben ser consistentes con el problema a resolver)
* **Contexto del desarrollo** (explique de manera breve en que consiste el desarrollo de su programa, señalando lo que permitirá resolver su programa).
* **Propuesta de solución** (software adjunto, no debe copiar todo el código, pero si mencionar que adjunto en la carpeta de entrega se puede revisar el código del programa).
* **Desarrollo** (contenido del informe, *vea ítem 5 para conocer el detalle).*
* **Reflexión final** (reflexión *individual* sobre el proceso de evaluación de scripts y la importancia de la seguridad en el desarrollo de software. Deben considerar cómo esta experiencia ha mejorado su comprensión de la programación segura y cómo aplicarán estos conocimientos en el futuro).

1. El desarrollo del informe debe considerar las siguientes temáticas:
   1. **Mitigación de Vulnerabilidades:**

Describir en qué medida el software garantiza la mitigación o reducción de vulnerabilidades en lo que respecta a la seguridad, es decir, debe indicar todo lo que implementó para que el procesamiento de los datos sea lo más seguro posible, acompañando la explicación con capturas puntuales del código en donde se implemente las intervenciones mencionadas.

* 1. **Buenas prácticas:**

Describir en qué momento hace uso de buenas prácticas durante el desarrollo de la solución, tanto a nivel de código, como a nivel de estructura, entre otros aspectos, acompañando la explicación con capturas del código que implemente las intervenciones mencionadas.

* 1. **Posibles amenazas:**

Describir posibles amenazas que atenten contra la estabilidad de su algoritmo cuando esté en ejecución, indicando, además los escenarios/contexto en los cuales podrían ocurrir o aparecer las amenazas mencionadas.

#COMENTARIO: AL SER USUARIOS Y CONTRASEÑAS TAN GENÉRICAS PUEDE SER MUY FACIL VIOLAR LA SEGURUIDAD DEL SISTEMA

1. **Recursos e instrumentos de evaluación del aprendizaje**

| Recursos de apoyo para las actividades | Instrumento de evaluación |
| --- | --- |
| Plantilla: Informe Metodologías de Desarrollo Seguro | Rubrica 4: Metodologías de Desarrollo Seguro |

# Referencias Bibliográficas

Arboledas Brihuega, D. (2017). *Criptografía sin secretos con Python.* RA-MA Editorial. Recuperado el 4 de Julio de 2024, de https://elibro.net/es/lc/inacap/titulos/106497

Cuevas Álvarez, A. (2016). *Python 3: curso práctico.* RA-MA Editorial. Recuperado el 4 de Julio de 2024, de https://elibro.net/es/lc/inacap/titulos/106404

Nolasco Valenzuela, J. S. (2018). *Python: aplicaciones prácticas.* RA-MA Editorial. Recuperado el 4 de Julio de 2024, de https://elibro.net/es/lc/inacap/titulos/106523

Ortega Candel, J. M. (2018). *Hacking ético con herramientas Python.* RA-MA Editorial. Recuperado el 4 de Julio de 2024, de https://elibro.net/es/lc/inacap/titulos/106513

Salit Rivera, J. D., López Jerez, M. Á., & Letelier Meza, J. C. (2024). *Introducción a la Programación Segura.* Rancagua: INACAP.