Evaluación # 4 (30%)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ÁREA ACADÉMICA** | | Tecnologías de Información y Ciberseguridad | | | **CARRERA** | Análisis de Sistemas | |
| **ASIGNATURA** | Programación Orientada a Objeto Seguro | | | | | **CÓDIGO** | TI3021/P13C1 y P13C2 |
| **SECCIÓN** | C1 | | |
| **SEDE** | Apoquindo | | | **DOCENTE** | Rebeca Cartagena Jessen | | |
| **Unidad de Aprendizaje** | | | N°4 | **Criterios a Evaluar** | |  | |
| **DURACIÓN** | Entrega 9-12-2024-21:00 hrs. | | | **FECHA** | | Semana última de noviembre | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **NOMBRE ALUMNO:** |  | | | | | |  | | |  |
|  | | | Apellido Paterno | | | Apellido Materno | | | Nombres | |
| **RUT:** | | | **-** | | | | | | | |
| **PUNTAJE MÁXIMO** | | 100 | | | **NOTA:** | | | Firma conforme | | |
| **PUNTAJE OBTENIDO** | |  | | |
| **Solicita re-corrección** | | **Sí** | | **No** | **Motivo:** | | | | | |

|  |
| --- |
| **Aprendizaje esperado**  1.1 Diseña software seguro en Python con conexión a base de datos, para solucionar una problemática, considerando la utilización de metodologías ágiles en la gestión tiempo y los roles del equipo |

|  |
| --- |
| **Criterios de evaluación**  4.1.1 Identifica requerimientos de la solución, de acuerdo con problemática.  4.1.2 Diseña diagramas de clases y casos de uso para la solución, de acuerdo con la notación UML.  4.1.4 Aplica metodologías ágiles de desarrollo, considerando los roles del equipo y los pasos de entrega de la solución.  4.1.5 Construye solución con autenticación segura (Hash) y conectada a base de datos según requerimientos identificados. |

**Instrucciones**

1. Esta evaluación puede ser desarrollada colaborativamente de 3 a 4 integrantes máximo.
2. Esta evaluación deberá ser subida al ambiente de aprendizaje de Inacap (AAI) en la fecha indicada
3. Dicha subida será en la actividad denominada ***evaluación 4***
4. No se acepta la evaluación # 4 por correo, por lo tanto, tome todas las precauciones para no dejar la entrega en los últimos 5 minutos del día de la entrega.

**Situación a desarrollar:**

Debe plantear el desarrollo de un caso de negocio para la implementación de una aplicación, utilizando conceptos de modelamiento de soluciones informáticas y metodología ágil en las actividades de apoyo al desarrollo de una aplicación utilizando una interfaz gráfica.

Debe desarrollar un software para una de las siguientes opciones:

A.- Agencia de viajes

B.- Clínica Veterinaria

C.- Pastelería

D.- Oficina de corretaje de propiedades

Se debe presentar un Informe, correspondiente a la Evaluación Sumativa 4, que consiste en el desarrollo de un software con interfaz gráfica desarrollado en Python utilizando el análisis de la aplicación, el uso de metodologías ágiles y modelamiento de soluciones informáticas, para el encargo de un cliente simulado.

Se entrega un Informe técnico con documentación que incluya diagrama UML, diagrama de clase, , producto backlog, el sprint backlog, y el Código del programa en Python con autenticación y almacenamiento en base de datos.

Los estudiantes de manera individual deberán realizar la defensa de la aplicación **realizando un código asociado a lo que han entregado. Lo que deberán hacer se los dirá su docente a cada integrante del grupo. La defensa es personal y tiene el espacio de 60 minutos para generar el desarrollo del código.**

**Cada estudiante recibirá de su docente lo que debe hacer a partir de lo que han entregado.**

**El informe debe contener:**

1.- Los requerimientos funcionales de un cliente ficticio en términos del sistema o software que debe desarrollar.

2.- Determinar un diagrama de clases / un caso de Uso general.

3.- Determine las historias de usuario y las tareas asignadas a cada historia de usuario relacionado con el desarrollo de software. Asigne una estimación en horas de desarrollo a cada HU.

4.- Establecer un plan de desarrollo que contemple el Product Backlog, un sprint backlog y que concluya con la entrega del proyecto desarollado en relación al software asignado. Asigne tareas específicas a cada integrante y estime tiempos. Realice reuniones para mejorar la entrega final (señale en que consistieron: Objetivos, tareas asignadas, recursos y tiempos)

5.- Indique el diseño de su base de datos y tablas para gestionar la solución

6.- Definir la interfaz de usuario a través de un mockup (lo puede realizar en Canva o en otro software gratuito)

7.- Presente un diagrama de componentes del sistema a desarrollar. Presente la organización de su sistema mediante el packard correspondiente.

**El Software a implementar debe contener:**

8.- Construcción en Python del sistema propuesto para resolver la problemática

9.- Interacción con la base de datos definida.

10.- Implementar la conexión a la base de datos y generar un CRUD correspondiente a la(s) tabla(s) definidas

11.- Debe al menos validar algunos datos y generar un proceso de autentificación de usuario al ingresar a la aplicación.

## Entregables Solicitados:

El equipo de estudiantes debe desarrollar un software en Python utilizando metodologías ágiles. Deben entregar el desarrollo de los siguientes items del informe técnico:

1. . Informe Técnico:

* Descripción de la problemática y los objetivos claros y precisos de la solución que se
* Patrones de Diseño Utilizados:
  + Identificación y explicación de los patrones de diseño aplicados en el sistema.
  + Razones detrás de la elección de estos patrones y cómo contribuyen a la calidad y mantenibilidad del código. (clases / Objetos / autentificación / encriptación)
* Planificación del Proyecto.
  + Product Backlog. (lista de actividades o tareas a realizar)
  + Sprint Backlog (que actividades realizara, al menos 2 sprint, uno para el informe técnico y otro para la codificación)

1. Documentación:

* Diagramas UML que representen la estructura del sistema, incluyendo diagramas de clases y casos de uso.
* Procesos de negocio detallados que describan cómo se gestionan los procesos del sistema que ud, resolverá.

1. Código del Programa en Python:

* El código debe estar bien comentado y seguir las mejores prácticas de programación.
* Se debe implementar un sistema de autenticación para la gestión de reservas.
* La información de la aplicación debe almacenarse en una base de datos, y se espera una descripción detallada del esquema de la base de datos.

**Ejemplo con Agencia de Viajes:**

## Requerimientos del Cliente:

La agencia de viajes tiene varios requisitos clave que desean que se aborden en el sistema:

1. ***Gestión de Destinos:***

* Cada destino tiene información como nombre, descripción, actividades disponibles y costo.
* Debe ser posible agregar, mostrar todos, modificar y eliminar destinos.

1. ***Paquetes Turísticos:***

* Los paquetes turísticos es un conjunto de servicios que se ofrecen a un turista a un precio único.
* Cada paquete tiene asociado el destino, hotel y pasajes por persona
* Cada paquete tiene un precio total que se calcula en función del destino seleccionado y la fecha de viaje.
* Los clientes pueden ver la disponibilidad de los paquetes para las fechas deseadas.

1. ***Reservas:***

* Los clientes pueden seleccionar y reservar paquetes turísticos disponibles.
* Se requiere un sistema de autenticación para que los clientes puedan realizar reservas y acceder a sus detalles personales.

Cada reserva debe registrarse en una base de datos para su posterior gestión como son las reservas, la actualización de destinos y la disponibilidad de paquetes turísticos.

1. **Actividades a desarrollar para generar su product backlog**

***Actividad 1: Recolección de Requisitos***

* Entender completamente los requisitos del cliente en términos de gestión de destinos, paquetes turísticos y reservas.
* Identificar cualquier requisito adicional y aclarar dudas con el cliente (docente).

***Actividad 2: Planificación del Proyecto***

* Establecer un plan de desarrollo basado en metodología ágil, dividiendo el trabajo en Sprint.
* Asignar tareas específicas a cada miembro del equipo.

***Actividad 3: Diseño Preliminar***

* Crear esbozos iniciales de la arquitectura del sistema.
* Diseñar la estructura de la base de datos.
* Definir la interfaz de usuario y las funcionalidades clave.

***Actividad 4: Desarrollo Iterativo***

* Implementar las funciones básicas de gestión de destinos y paquetes turísticos en la primera iteración.
* Introducir la autenticación y la gestión de reservas en iteraciones posteriores.
* Realizar pruebas de unidad continuas para garantizar la calidad del código.

***Actividad 5: Documentación y mejora Continua***

* Mantener actualizada la documentación, incluyendo diagramas UML y procesos de negocio, a medida que se desarrolla el software.
* Asegurarse de que la documentación refleje con precisión el estado actual del sistema.

***Actividad 6: Entrega Final***

* Entrega del informe técnico
* Presentar la solución
* Defensa individual del trabajo: Consiste en realizar una modificación a la aplicación a partir de lo que docente le indique a cada grupo. El tiempo para realizar esta actividad evaluada es de 70 minutos.

De acuerdo a lo realizado recibirá la nota de su evaluación.

La nota se divide en dos partes:

1. Informe técnico: 30 puntos
2. Entrega del sistema que debe ejecutarse para optar a los 70 puntos, en caso que no se ejecute, se revisa el código solo para obtener un máximo de 30 puntos.

Fecha entrega lunes 9-12-2024 antes de las 21:00 hrs.

Fecha Defensa individual: miércoles 11-12-2024 en hora de clases el estudiante se presenta y debe realizar alguna mejora, cambio o actualización de alguna funcionalidad de su sistema que la docente indique.

**Historia de Usuario**

**Yo como [perfil de usuario], quiero [objetivo del software para lograr [propósito]**

|  |  |  |
| --- | --- | --- |
| **Historia de usuario** | **Tareas** | **Total horas** |
| **Como usuario** de la aplicación **quiero poder ingresar** de manera segura y confiable mi nombre de usuario o rut y contraseña **para proteger mis datos y acceder a** las funcionalidades de la aplicación**.** | **Crear la ventana de inicio con campos para ingresar nombre de usuario o rut y contraseña**  **Incluir botones de inicio de sesión, registrar y recuperar contraseña**  **Añadir un indicador de carga para cuando el usuario haga clik en “iniciar sesión”**  Validar los campos de usuario y contraseña, que no estén vacíos antes de enviarlos  Verificar que el nombre de usuario o Rut sea correcto  Comprobar que la contraseña cumpla con los estándares minimos de seguridad  Almacenar las contraseñas encriptadas.  Comparar las credenciales ingresadas con las que están almacenadas en la base de datos  Manejar casos de error como usuario o contraseña incorrecta o cuenta bloqueada(si se establece un límite de intentos) | **2 horas**  **1 hora**  **0,5 horas**  **0,5 horas**  **0,5 horas**  **1 hora**  **1 hora**  **0,5 horas**  **1 hora** |
| **Como usuario** **quiero una interfaz** limpia y sencilla **para ingresar** sin dificultad a la aplicación. | Realizar un mockup de la aplicación, correspondiente a la página principal  Ajustar la planilla utilizando las funcionalidades del tkinter  Crear un logo  Incorporar los colores al diseño de la página principal | **2 horas**  **2 horas**  **1 hora**  **1 hora** |
| **Como usuario** de la aplicación **quiero poder gestionar los registros** en el sistema través de GUI **para lograr crear, ver, actualizar y eliminar** registros de manera eficiente | Definir la estructura de la base de datos  Identificar los campos y atributo necesarios para cada tabla.  Crear una GUI o un formulario para visualizar los registros de la tabla  Incluir los botones de Crear, editar, eliminar y actualizar registros  Desarrollar el código para abrir el formulario de creación al hacer click en “**Crear “o “Nuevo”**  Implementar el código para validar los datos antes de almacenar el registro en la tabla  Guardar el nuevo registro en la base de datos  Configurar la tabla o lista para mostrar los registros almacenados.  Crear un método **que recupere** y **muestre** los datos de la base de datos.  Añadir la opción de ver un registro especifico desde una ventana emergente.  Habilitar la opción de seleccionar un registro y hacer **click en botón Editar.**  Abrir el formulario de edición  Rellenar el formulario  Validar los datos ingresados.  Guardar los cambios en la base de datos y actualizar la tabla de datos para visualizar los cambios**.**  Permitir que el usuario seleccione un registro y haga **click en botón eliminar,** para eliminar un registro.  Mostrar una alerta o confirmación antes de eliminar el registro para evitar eliminaciones accidentales.  Remover el registro de la base de datos y actualizar la tabla para reflejar los cambios.  Realizar las pruebas unitarias de cada función del mantenedor.  Probar que cada función opere correctamente.  Configurar permisos para que solo ciertos usuarios puedan realizar las funciones de editar y eliminar registros. |  |

**Evaluación del informe técnico – 30 puntos (escala de apreciación).**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Puntos para evaluar** | **Indicadores** | ***Excelente (3 puntos)*** | ***Suficiente (2 puntos)*** | ***insuficiente (1 punto)*** | ***No realizado (0 Ptos.)*** |
| Aplicación de la metodología Ágil en el desarrollo de la aplicación a realizar utilizando Python interaccionando con base de datos | El objetivo general y específicos son claros y precisos en relación a la aplicación abordada. |  |  |  |  |
| Establece de manera clara los requerimientos funcionales de la aplicación |  |  |  |  |
| Identifica clases y objetos a utilizar así como la organización del packard |  |  |  |  |
| Diseña la estructura de la base de datos |  |  |  |  |
| Define el producto backlog en relación a todas las actividades que debe desarrollar. |  |  |  |  |
| Establece las actividades asociadas a las historias de usuarios del product backlog |  |  |  |  |
| Asgna tareas específicas a cada miembro del equipo |  |  |  |  |
| Diseña la interfaz gráfica |  |  |  |  |
| Señala la función y dinámica de cada reunión ceremonial. |  |  |  |  |
| Entrega una conclusión del trabajo realizado. | Identifica las fortalezas logradas en su desarrollo e identifica las oportunidades de mejora. |  |  |  |  |

**Evaluación del código referente a la aplicación – 70 puntos (escala de apreciación).**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Puntos para evaluar** | **Indicadores** | ***Excelente (7 puntos)*** | ***Bueno (5 puntos)*** | ***Suficiente (3 puntos)*** | ***insuficiente (1 punto)*** | ***No realizado (0 Ptos.)*** | ***Ptje. Máximo*** |
| Interfaz gráfica | La interfaz es clara, precisa y resuelve la problemática planteada |  |  |  |  |  |  |
| Validación | Realiza validación de al menos los datos relevantes |  |  |  |  |  |  |
| Try - Except | Captura errores con la finalidad de lograr que el sistema no deje de ejecutarse |  |  |  |  |  |  |
| Herencia | Hay al menos una clase que utiliza herencia |  |  |  |  |  |  |
| Aplicación. Mantenedor | Organización de la carpetas tipo packard |  |  |  |  |  |  |
| Las bases de datos y tablas creadas son pertinentes para la ejecución de la aplicación |  |  |  |  |  |  |
| El mantenedor realiza las funciones del CRUD |  |  |  |  |  |  |
| El menú presentado es acorde a la APP . |  |  |  |  |  |  |
| Las clases definidas son acorde en atributos y creación de objetos |  |  |  |  |  |  |
| Autentificación de usuario | El sistema parte solo si el usuario se autentifica con contraseña encriptada |  |  |  |  |  |  |