

- Cada pregunta **múltiple opción** contestada correctamente tiene un valor de 2 puntos. Cada pregunta incorrecta de la múltiple opción resta 0,66 puntos.
- La evaluación es de carácter individual y la duración es de 2 horas.
- **El puntaje total del parcial es de 44 puntos.**

Múltiple Opción

1. Sobre la ingeniería de software
 - a) Los métodos, herramientas y técnicas a utilizar en el desarrollo de software dependen de varios factores dentro de los cuales el tipo de software a construir es quizás el más significativo
 - b) a) y un tipo de software son las aplicaciones independientes, que son aquellas aplicaciones que aunque funcionan en red utilizan protocolos y tecnologías propietarias, y por tanto, independientes.
 - c) **a) aunque los límites de los tipos de software muchas veces son borrosos y una aplicación puede corresponder a más de un tipo.**
 - d) Ninguna de las anteriores es correcta.
2. En el área de ética profesional existen algunas cuestiones de responsabilidad profesional más allá del aspecto legal. Sobre estas cuestiones, indique la opción incorrecta.
 - a) **Confidencialidad - implica respetar la confidencialidad de los empleadores o clientes, siempre que se haya suscrito un acuerdo de confidencialidad.**
 - b) Competencia - implica no aceptar trabajos que están fuera del nivel de competencia.
 - c) Derechos de propiedad intelectual - implica conocer las leyes locales que rigen el uso de la propiedad intelectual, como patentes, derechos de autor, etc.
 - d) Mal uso de la computadora – implica no usar mal las computadoras de otras personas.
3. El cambio es inevitable en todo proyecto de software grande. Con respecto a esto:
 - a) Sea cual sea el proceso de desarrollo usado, es importante que éste pueda hacer frente a los cambios que surjan.
 - b) **(a) y el cambio generalmente implica retrabajo. En el curso vimos 2 enfoques para reducir los costos del retrabajo: anticiparse al cambio o tolerar los cambios.**
 - c) El enfoque de “Tolerar los cambios” implica incluir actividades en el proceso que permitan anticiparse o predecir posibles cambios antes de que sea necesario hacer un retrabajo significativo. Por ejemplo, hacer prototipos para refinar los requerimientos.
 - d) (c) y una de las ventajas del desarrollo incremental es que como los servicios más importantes se desarrollan primero, éstos son los que reciben más testing, ya que luego se los sigue probando (por ejemplo, cuando se integran con otros servicios).
4. En relación a las prácticas utilizadas por los procesos ágiles:
 - I. Una de las formas de representar escenarios de uso de un sistema es a través de las historias de usuario. Estas se pueden utilizar en la planificación de las iteraciones del sistema.
 - II. Los siguientes son ejemplos de refactorización: agregar una jerarquía entre clases para evitar código duplicado, armar rutinas para encapsular una lógica común y llamar a las mismas en lugar de duplicar el código.
 - III. En el desarrollo dirigido por pruebas se escriben los casos de prueba antes de el código. Dado que se tiene que entender bien la especificación para escribir las pruebas, por lo general sucede que las dudas y omisiones se clarifican antes de empezar con el desarrollo.
 - IV. En la programación por pares, los programadores trabajan de a pares para desarrollar el software. Para que sea más eficiente el trabajo, se aconseja trabajar siempre que sea posible con el mismo par.
 - V. El prototipado de software es una técnica que surge a partir de la creación de los métodos ágiles.

Seleccione la opción correcta:

 - a) Todas las opciones son correctas.
 - b) Sólo (I), (III) y (IV) son correctas.
 - c) Sólo (II) y (III) son correctas.
 - d) **Sólo (I), (II), (III) son correctas.**

5. Considere las siguientes afirmaciones relacionadas a la mejora de procesos:
- I. La mejora de procesos implica conocer los procesos existentes y cambiar esos procesos para mejorar la calidad del producto y/o reducir los costos del desarrollo y el tiempo.
 - II. Dos enfoques distintos para la mejora de procesos son: el enfoque de madurez de procesos y el enfoque ágil.
 - III. Las reuniones de retrospectiva sólo sirven para la mejora de procesos ágiles.
 - IV. Para poder evaluar la mejora del proceso se necesita recolectar datos. La medición del proceso implica medir uno o más atributos del proceso/producto de software. Siempre que sea posible se deben recolectar datos cuantitativos. Luego, las medidas recolectadas ayudarán a decidir si las mejoras implantadas han sido efectivas.
 - V. CMMI, el modelo de capacidad de madurez del SEI es un modelo que sigue el enfoque de madurez de procesos. Éste cuenta con 5 niveles de madurez.

Seleccione la opción correcta:

- a) Todas las opciones son correctas.
- b) Sólo (I), (II), (IV) y (V) son correctas.
- c) Sólo 3 opciones son correctas.
- d) Sólo (I), (II), (III) y (IV) son correctas.

Dada la siguiente realidad planteada:

Imagine que trabaja en una empresa de software, la cual ha ganado una licitación del estado, por lo cual uno de sus clientes es la Intendencia de Montevideo (IM), en particular, el Sistema de Transporte Metropolitano (STM). Se quiere contar con un sistema de transporte metropolitano, en donde se utilice la huella dactilar (en vez de la tarjeta STM) para el pago de los boletos de estudiantes. En la licitación, se espera que el desarrollo del sistema cumpla con los siguientes hitos:

- Especificación de requisitos validada a los 3 meses de comenzado el proyecto.
- Prototipo funcional para validar requisitos a los 6 meses de comenzado el proyecto.
- Versión funcional para plan piloto al año de comenzado el proyecto, el cual servirá para la validación del sistema.
- Versión final para colocar en producción luego de 6 meses de piloto.

6. Dada la realidad planteada y los modelos de proceso vistos en el curso, indique cuál afirmación es correcta:
- I. Una metodología que puede aplicar a este proyecto es Scrum puro, ya que se necesita un prototipo para validar a los 6 meses, práctica que se utiliza y es recomendada en dicha metodología.
 - II. Debido a que el documento de requisitos tiene que estar validado y completo a los 3 meses, la metodología que mejor aplica en este caso es “cascada”.
 - III. Si se conocieran datos del equipo de desarrollo se podría identificar a Scrum puro como una buena metodología a aplicar.
 - IV. La existencia de un contrato indica que el proceso a seguir tiene que tener un mínimo de desarrollo dirigido por planes, para asegurarse que los hitos planteados se cumplan de forma exitosa.
 - V. Debido a que ya existe un sistema basado en las tarjetas STM y que existe en el mercado software para la lectura de huellas dactilares, se podrían tener en cuenta algunas de las etapas sugeridas por el enfoque de integración y configuración como parte del proceso de desarrollo.

Seleccione la opción correcta:

- a) Sólo (II) y (IV) son correctas
- b) Sólo (III) y (V) son correctas
- c) Sólo (III) es correcta
- d) Sólo (IV) y (V) son correctas

7. En base a la realidad anterior, suponga que usted estuvo como responsable de la implementación del prototipo relacionado al primer hito del contrato con la IM. Luego de su presentación, el responsable del proyecto estuvo tan encantado con el funcionamiento que le sugirió implantar el prototipo en producción y adelantar la fase piloto con el mismo. Indique cuáles de las siguientes afirmaciones/consideraciones tomaría en cuenta:
- Si el prototipo fue realizado con una arquitectura no escalable y sin haber realizado pruebas de desempeño: accedería a la propuesta del cliente, pero con la advertencia de que habría que actualizar dicha versión rápidamente (por el riesgo que implica) y eso va a costar un dinero extra al presupuesto adjudicado en la licitación
 - Si el prototipo dista mucho de la versión final, no sería posible implantarlo en producción para adelantar la fase piloto, ya que el mismo va a cambiar (o será desechado) y las validaciones que puedan haberse hecho en esa fase no tendrán validez alguna, por lo cual habría que realizar otra fase piloto posteriormente
 - Realizar una validación temprana del sistema con un prototipo en producción es siempre bienvenida (feedback del cliente y usuarios de forma temprana)
 - El diseño de este prototipo fue una estrategia de anticipación al cambio, justamente teniendo como objetivo validar los requisitos del cliente. El mismo ha cumplido su función tal cual se estipulaba en el contrato.
 - Desaconsejaría su puesta en producción hasta que las funcionalidades (o las que se deseen implantar) no estén completamente desarrolladas, los requisitos no funcionales adecuadamente chequeados y que el mismo haya pasado las pruebas necesarias.

Seleccione la opción correcta:

- Sólo las consideraciones I, II y V son correctas
 - Sólo la consideración III es correcta
 - Sólo las consideraciones IV y V son correctas
 - Sólo las consideraciones II, IV y V son correctas**
8. Durante la realización del proyecto, el equipo de desarrollo decidió incluir en su proceso los siguientes elementos: reuniones de revisión (retrospectiva), integración continua y test-first development. Seleccione la opción correcta:
- Es claro que el equipo decidió utilizar Scrum como metodología de desarrollo.
 - Es claro que el proceso era iterativo incremental, para poder realizar las reuniones de revisión al final de cada iteración.
 - Es claro que al introducir test-first development es necesario introducir integración continua, ya que se reutilizan los casos de prueba generados por test-first development al momento de validar la integración de un módulo modificado a la versión actual del sistema.
 - Ninguna de las opciones es correcta.**

-
9. Con respecto a la construcción de software:

- La elección del modelo de desarrollo a utilizar no afecta la construcción.
- Cierta cantidad del trabajo de diseño se realiza en construcción. A su vez, tanto las pruebas unitarias como parte de las pruebas de integración se realizan durante la construcción.
- La anticipación al cambio afecta como el software es construido. En este sentido, un ejemplo es guardar los estados posibles de las facturas en un tabla de la base de datos o en un archivo de configuración en lugar de ponerlos fijos en el código.
- La reutilización, por lo general, permite reducir los costos de la documentación aunque no del desarrollo.
- Las APIs tienen como ventaja que disminuyen el acoplamiento.

Seleccione la opción correcta:

- Todas son correctas.
- Sólo 2 opciones son correctas.
- Sólo (I), (II) y (III) son correctas.
- Sólo (II), (III) y (V) son correctas.**

Considere la siguiente realidad planteada:

Con el objetivo de la compra de libros entre estudiantes de ingeniería, la facultad promueve una nueva aplicación web.

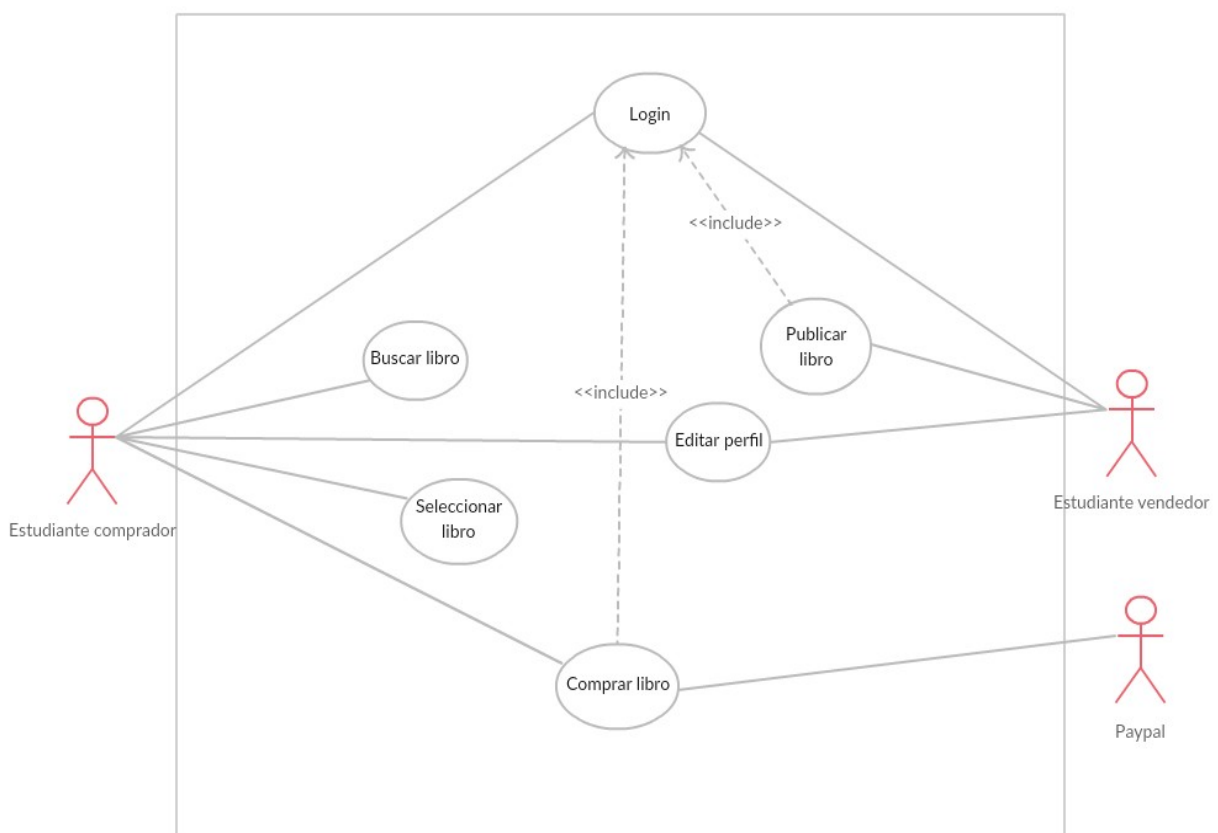
El estudiante puede elegir entre publicar libros, para que otros estudiantes puedan comprarlo, o comprar libros.

Si el estudiante quiere comprar un libro, se despliegan filtros de búsqueda: título o autor del libro. Al utilizar los filtros, la aplicación muestra los libros que cumplen con los criterios seleccionados. El usuario podrá seleccionar un libro, donde se mostrarán los detalles del mismo, incluyendo los datos del estudiante que vende el libro, y una opción para poder comprarlo. El usuario podrá comprar el libro solo si está logueado en el sistema. El sistema permite acordar con el vendedor la forma de pago o pagar a través de PayPal.

Si el estudiante quiere publicar un libro, deberá estar logueado en el sistema e ingresar los datos del libro que quiere vender: título, autor del libro, fecha de publicación y opcionalmente una descripción del mismo.

El estudiante podrá ingresar a la aplicación con el usuario y contraseña otorgado por bedelia al momento de inscribirse a una carrera.

A continuación se muestra la primer versión del diagrama de casos de uso, el cual todavía no está terminado.



Respecto a la realidad planteada y el diagrama de casos de uso, conteste las siguientes preguntas:

10. Respecto al diagrama de casos de uso:

- I. El caso de uso "Seleccionar libro" podría extender al caso de uso "Buscar libro".
- II. El caso de uso "Comprar libro" podría extender al caso de uso "Seleccionar libro".
- III. El caso de uso "Seleccionar libro" podría incluir al caso de uso "Publicar libro".
- IV. El caso de uso "Editar perfil" podría incluir al caso de uso "Login".

Seleccione la opción correcta:

- a) Sólo (I), (III) (IV) son correcta.
- b) Sólo (II), (III) (IV) son correcta.
- c) **Sólo (I), (II) y (IV) son correctas.**
- d) Todas son correctas.

11. Respecto a los actores del sistema, seleccione la opción correcta:
- Dos actores del caso de uso “Comprar libro” son: Estudiante comprador y Paypal.
 - El actor Paypal no debería de considerarse como un actor en el caso de uso “Comprar libro” dado que el estudiante comprador puede acordar con el estudiante vendedor la forma de pago. La solución más adecuada es crear un nuevo caso de uso “Comprar libro por Paypal”.
 - Paypal no es un actor dado que es un sistema externo.
 - En el diagrama de casos de uso presentado anteriormente están presentes todos los actores del sistema.
12. Con respecto al caso de uso “Publicar libro”:
- Una vez realizado el login, el caso de uso comienza con la acción del usuario que selecciona la opción correspondiente a publicar libro.
 - El caso de uso comienza con el sistema mostrando la opción de comprar o publicar un libro.
 - Los únicos flujos alternativos en el caso de uso son: flujos alternativos generales y el flujo alternativo correspondiente a la validez del campo “fecha de publicación” (como por ejemplo: formato válido).
 - (c) y el flujo alternativo correspondiente a la validez de la fecha de publicación debe volver a algún paso del flujo normal.

13. La empresa donde trabaja fue encargada del desarrollo de la aplicación “FingFeedly” (presentada en el Obligatorio 1). Si bien hay varios requisitos relevados y priorizados, el cliente manifestó que tiene algunas ideas para incorporar a la aplicación, pero que todavía no están definidas. Por este motivo y porque la empresa tiene experiencia en metodologías ágiles, se decidió seguir Scrum para el desarrollo de la aplicación. El equipo de desarrollo definió un conjunto de características que una historia de usuario debe cumplir para poder comprometerse a su entrega (*Definition of Ready*). Para el primer sprint el equipo se comprometió a desarrollar las siguientes historias de usuario:

- Buscar feed: Como usuario administrador quiero poder buscar todos los feeds disponibles en la web para poder habilitar y deshabilitar los feeds.
- Habilitar y deshabilitar feed: Como usuario administrador quiero poder habilitar y deshabilitar los feeds para que los mismos queden disponibles (o no) para suscripción por los usuarios de la aplicación, con el fin de estar al tanto de las noticias de su interés.
- Suscribirse y desuscribirse a un feed: Como usuario de la aplicación (estudiante y administrador) quiero poder suscribirme a un feed para poder estar informado de las últimas noticias del mismo así como también desuscribirme cuando ya no me interesan sus noticias.

Luego de una semana de comienzo del primer sprint, el equipo de desarrollo está atrasado a causa de que ningún integrante tiene experiencia en cómo consumir los feeds.

- En la planificación del primer Sprint, el equipo debió de haber agregado una historia de usuario de tipo SPIKE para investigar el consumo de feeds.
- Agregar en la planificación del primer sprint una historia de usuario de tipo SPIKE no hubiera sido una solución, dado que ya existen otras aplicaciones que consumen feeds en el mercado y por lo tanto no hay que hacer un estudio de viabilidad.
- El equipo deberá evaluar si las historias de usuarios cumplían con el conjunto de características definidas (*Definition of Ready*) o ajustar este conjunto.
- Dado que el consumo de feeds es un requisito no funcional, es probable que no se hayan especificado de forma detallada los criterios de aceptación.
- El atraso respecto a la planificación del primer sprint es aceptable, dada la poca experiencia del equipo de desarrollo en consumir un feed.

Seleccione la opción correcta:

- Solo las afirmaciones (I) y (III) son correctas
 - Solo las afirmaciones (II) y (III) son correctas
 - Solo las afirmaciones (II) y (V) son correctas
 - Solo la afirmación (IV) es correcta.
14. Sobre los prototipos:
- Las pruebas de concepto son prototipos verticales, esto significa que implementan una porción funcional de la aplicación.
 - a) y permiten resolver incertidumbres sobre la factibilidad de la arquitectura propuesta u otros riesgos técnicos.
 - Las pruebas de concepto son prototipos horizontales, esto significa que se enfocan en porciones de la interfaz de usuario
 - c) aunque no permiten explorar comportamientos específicos del producto.

15. La técnica de entrevistas individuales para la obtención de requisitos:
- Tiene como ventaja que es una técnica orientada a las personas y es flexible.
 - Es efectiva para obtener información respecto a los requisitos y restricciones de la organización.
 - a) aunque no es muy útil en caso de tener cientos de usuarios que entrevistar para relevar los requisitos.
 - Todas las anteriores son correctas**
16. Sobre el Diseño de la Arquitectura de Software:
- Es un proceso creativo por lo que el proceso puede variar según el sistema que va a ser desarrollado**
 - a) y las restricciones de un proyecto suelen ser críticas para la definición de la arquitectura.
 - b) y la arquitectura de un sistema puede ser diseñada en torno a un patrón de arquitectura o un “estilo”, pero no ambos,
 - Es una etapa temprana del proceso de diseño del sistema, y debe llevarse a cabo en paralelo con actividades de especificación de requisitos.
17. Con respecto a la arquitectura de software:
- Diseñar y documentar explícitamente la arquitectura favorece la comunicación entre stakeholders.**
 - El estilo arquitectónico a elegir depende más fuertemente de los requisitos funcionales que de los no funcionales.
 - El estilo arquitectónico “shared-data” (o repositorio) tiene como desventaja que resulta en una forma ineficiente de compartir grandes cantidades de datos.
 - Dos opciones son correctas,
18. Acerca de la arquitectura y diseño puede decirse que:
- Es una descripción de la organización de un sistema de software.
 - a) y algunas propiedades como la usabilidad de la interfaz de usuario, la disponibilidad, el rendimiento y la seguridad dependen de la arquitectura utilizada.
 - a) y algunos patrones comunes son: arquitectura en capas, cliente-servidor, repositorio.**
 - La mejor forma de documentar la arquitectura es mediante una vista de despliegue (vista física).
19. Indique cuáles de las siguientes son características de los sistemas distribuidos:
- Compartir recursos - implica compartir recursos de hardware y software.
 - Apertura - implica que el código sea abierto para proveedores y clientes.
 - Concurrencia - implica procesamiento concurrente para mejorar rendimiento.
 - Escalabilidad - implica mayor rendimiento al agregar nuevos programadores.
 - Tolerancia a fallas - implica la capacidad de continuar en funcionamiento después de que se ha producido un error.
- I, III y V son correctas.**
 - Ninguna de las afirmaciones es correcta.
 - I, II, III y V son correctas.
 - Todas las afirmaciones son correctas.
20. Sobre el modelado del sistema:
- Se recomienda que los modelos cubran todo el sistema, de modo de cumplir con que los requisitos sean completos.
 - Los modelos sirven tanto para elaborar y explorar requisitos así como para diseñar soluciones de software.**
 - a) y b) son correctas.
 - Todas las anteriores son incorrectas.
21. Definir la arquitectura de un sistema implica:
- Tener claro los requisitos y el patrón de diseño a utilizar.
 - Definir los componentes del sistema, sus interfaces y reglas de interacción.**
 - Dividir en módulos simples y comenzar la implementación.
 - Todas las anteriores son correctas.
22. Sobre construcción de software:
- No es solo codificación, también implica, por ejemplo, pruebas unitarias y debugging.**
 - Se busca minimizar la complejidad, aplicando técnicas de programación defensiva ante cambios.
 - Las pruebas del sistema forman parte de la construcción, a través de las cuales se valida lo codificado.
 - Dos opciones son válidas.