# **Finales**

## **25/07/2016**

1. Testing: ¿Que es el Pass Around?

**RTA**:

* Es la técnica de Descheck pero con múltiples participantes. Esta técnica implica solicitar una revisión del código a más de un colaborador para verificar potenciales errores.
* El pass around se encarga de mitigar el potencial error de que solo una persona verifique lo que se hizo

1. Diferencia entre esfuerzo y duración:

**RTA**:

* Esfuerzo son la cantidad de horas/hombre que le lleva a una persona con conocimientos realizar una tarea de forma ininterrumpida.
* La duración es el esfuerzo estimado previamente, pero agregandole los feriados, fines de semana, rotación entre tareas, etc.

## **10/07/2017**

1. ¿Cuando se puede agregar gente a un proyecto?

**RTA**:

* Jamas se tiene que agregar gente a un proyecto ya retrasado.
* Solo se puede agregar si las tareas son independientes entre sí y no requieren comunicación con los otros devs ni capacitación

1. Según las visiones de la calidad, mayor calidad siempre implica un mayor costo.

**RTA**:

* Falso. Se puede aumentar la calidad sin aumentar el costo.
  + Por ejemplo, utilizando un diseño orientado a pruebas para reducir el costo de las mismas.
* A su vez, la visión de la manufactura implica la reducción de residuos en cada etapa, lo que implica minimizar los costos para agrandar la calidad.

1. FP son independientes de la tecnología.

RTA:

* Verdadero porque se encarga de identificar las funcionalidades más importante y sus parámetros de entrada y salida, sin importar como está implementado.

1. Criterios para identificar IC:

**RTA**:

* Componentes que van a ser compartidos por 2 o más equipos, ya que se tiene que guardar el historial de que y cuando cambió cada uno.
* Componentes que son críticos para el proyecto, para corroborar su integridad.
* Componentes que sufrirán cambios.
* Componentes interdependientes entre sí.

1. Definición de complejidad ciclomática.

**RTA**:

* Es una métrica de SW cuantitativa para medir la complejidad lógica de un programa

1. Definir clases de equivalencia inválidas, condiciones de borde y conjetura de errores para “Edad[18 - 65]

**RTA**:

* Inválidas: [0 - 17] U [66 - 120]
* Borde: [18, 65]
* Conjetura de errores: ¿Que pasa si se ingresa una edad negativa? ¿Que pasa si se ingresa una edad con coma?

1. Testing: Si hay un resultado obtenido distinto del esperado hay un defecto.

**RTA**:

* Falso. Si diferen los resultados esperados y obtenidos se tiene un incidente. Este puede deberse a problemas del entorno, de la prueba o del usuario que la realiza. En el caso de que todo esto se valide y no sea así, entonces se determina que es una falla en el sistema y puede deberse a uno o más defectos presentes.

## **24/07/2017**

1. Definición de driver.

**RTA**:

* Es un requerimiento vital para el proyecto. El mismo tiene una flexibilidad baja.

1. Preguntas que hay que realizarse en la etapa de Status & Accounting.

**RTA**:

* ¿Quien hizo el cambio?
* ¿Quien solicito el cambio?
* ¿Cuando se realizó el cambio?
* ¿En que estado está el cambio?
* ¿Que implica el cambio?

1. ¿Es eficiente, a partir de condiciones de prueba, derivar clases de equivalencia?

**RTA**:

* Si, ya que las condiciones de prueba son descripciones de las situaciones que se quieren probar y ante las que el sistema debe dar respuesta. Las clases de equivalencia ayudarían a, con un solo ejemplo de cada tipo, lograr cubrir dichas pruebas con el menor esfuerzo posible.

## **11/12/2017**

1. Situaciones donde NO usar Timebox Development

**RTA**:

* Se tiene un equipo grande (más de 10 participantes)
* Se tiene un proyecto de mediana o largo duración (Más de 3 meses)
* No se está dispuesto a sacrificar requerimientos por sobre la calidad.

1. Use Case Points: Clasificación de CU:

**RTA**:

* Simple: 3 o menos transacciones.
* Medio: 4 a 7 transacciones.
* Complejo: Más de 7 transacciones.

1. Ventajas Wideband Delphi:

**RTA**:

* Elimina el problema de tener una sola persona en el equipo.
* Genera un sentido de propiedad ya que todos los integrantes opinan y son escuchados.
* Se puede utilizar en etapas tempranas de un proyecto.
* Se puede utilizar en proyectos sin historia.
* Facil de implementar.

1. Ventajas Inspección por sobre Testing:

**RTA**:

* Encuentra malas prácticas de programación.
* Encuentra defectos de forma temprana antes de que se transformen en fallas.
* Encuentra defectos que el testing no podría encontrar.

## **18/12/2017**

1. Requisitos para aplicar Timebox Development.

**RTA**:

* La metodología debe ser aplicable en el tipo de proyecto.
* Se debe estar dispuesto a sacrificar requerimientos en vez de calidad para cumplir con el tiempo estipulado.
* Se requiere identificar las funcionalidades principales a desarrollar.
* Se requiere la participación del usuario final.

1. ¿Cual es el rol de Status & Accounting en SCM?

**RTA**:

* Gestión y control de los ICs que se encuentra gestionados bajo la configuración. Es importante que se tenga conocimiento, en todo momento, de los ICs aprobados, el estado de los cambios solicitados y se genere un reporte de trazabilidad sobre la línea de base.

1. Ventajas de Wideband Delphi:

**RTA**:

* Se define como juicio experto en grupos, por lo que se elimina el problema de que una sola persona aporte todo el conocimiento.
* Es facil de implementar.
* Se puede utilizar en proyectos sin historia.
* Se puede aplicar en etapas tempranas de un proyecto.
* Genera un sentido de propiedad en los participantes ya que todos aportan para definir la estimación.

1. Describir los tipos de auditoría.

**RTA**:

* Funcional: Se encarga de validar que lo efectivamente entregado sea lo pactado previamente.
* Física: Encargada de verificar la estructura.
  + Además se encarga de validar que el código subido coincida con el cambio solicitado.
  + También analiza que no se haya subido otro código que no haya sido solicitado.
* De proceso: Encargada de verificar el cumplimiento del proceso SCM.

1. ¿Que implica que el testing sea eficaz y eficiente?

**RTA**:

* Eficiente: Sea lo más rápido y barato posible.
* Eficaz: Encuentre la mayor cantidad de errores y los más importantes.
  + No detecte falsos positivos.

1. Ventajas de Inspección por sobre Testing.

**RTA**:

* Encontrar fallas en etapas tempranas.
* Encontrar malas prácticas.
* Encontrar defectos que el testing no puede encontrar.
* La inspección es un proceso formal y detallado donde todos los participantes tienen roles asignados.
  + Hay un director que guía el proceso, el cual no debe ser el autor de la pieza a revisar.
  + Debe haber un presentador.
  + El resto del equipo se encarga de registrar los problemas.

## **05/02/2018**

1. Mencionar las 5 visiones de calidad.

**RTA**:

* **Trascendental**: La calidad se puede reconocer, pero no se puede definir.
  + Está relacionada con la percepción de los individuos.
  + Es muy utilizada en Marketing.
  + Es subjetiva.
* **De usuario**: Adecuación al propósito.
  + Satisfacer necesidades y ausencia de defectos.
  + Es subjetiva.
* **Basada en valor:** Lo que los individuos estan dispuestos a pagar por el producto.
  + Precio acorde dentro de la oferta y la demanda
  + A mayor precio, se percibe una menor calidad.
  + Es subjetiva.
* De manufactura: Conformidad con las especificaciones.
  + Se establece un proceso de buenas prácticas para lograr un acuerdo sobre la calidad.
  + Es objetivo porque se establecen parámetros fijados para todos.
* Del producto: Vinculado a las características propias del producto.
  + Relacionado con la ISO 25010.
  + Es objetivo ya que evalua según ciertos parámetros.

1. El SCCB está compuesto por perfiles del tipo \_\_\_\_\_\_\_\_\_ y su función es \_\_\_\_\_\_\_\_.

**RTA:**

* Técnico (Líder, PO, analista)
* Aprobar o rechazar los cambios a la configuración.

1. Function Points: los 5 elementos que se utilizan.

**RTA**:

* ALI: Archivos lógicos internos.
* AIE: Archivos interfaz externa.
* EE: Entradas externas.
* SE: Salidas externas.
* CE: Consultas externas.

1. 3 tipos de equipos (nuevo que no se conocen, uno que está teniendo mal desempeño, y otro que trabaja hace mucho y tienen experiencia) e indicar qué tipo de estimación utilizarías para cada uno (Paramétrica, Rudimentaria, Indistinta)

**RTA**:

* Trabajan hace mucho: Paramétricas ya que tienen historia con la que pueden contar para refinar las estimaciones.
* Nuevos y no se conocen: No paramétricas. No tienen historia disponible y todavía no tienen conocimiento de su equipo de trabajo.
* Mal desempeño: Indistinto.

1. V/F. Una métrica por sí sola sirve para realizar toma de decisiones.

**RTA**:

* Falso.
* Una métrica no es más que indiciador cuantitativo de un atributo. Para que sea utilizado en toma de decisiones, debe estar acompañado de un contexto y un objetivo.

## **05/08/2022**

1. Asociar subatributos de la 25010.

**RTA**:

* Adecuación Funcional: Pertinencia Funcional.
* Usabilidad: Operabilidad.
* Eficiencia de desempeño: Capacidad.
* Fiabilidad: Disponibilidad.
* Mantenibilidad: Reusabilidad.

1. Timebox: Las funcionalidades son un grado de libertad.

**RTA**:

* Verdadero. El tiempo es la única constraint. Se pueden relegar funcionalidades para cumplir con el tiempo y no disminuir la calidad.

1. Cynefin: Contexto complejo y caótico estan desordenados, el camino a seguir se determina según patrones emergentes.

**RTA**:

* Verdadero.

1. Kanban: El WIP indica la cantidad de items paralelos durante todo el proceso.

**RTA**:

* Falso. Indica los items que se pueden desarrollar en paralelo en cada etapa del proceso.

1. Prácticas asociadas a la Integración continua.

**RTA**:

* Uso de script para automatizar los procesos.
* Uso de un único repositorio para simplificar el acceso y la integridad.
* Uso de la misma herramienta a lo largo del proyecto para eliminar potenciales errores que no sean parte del sistema.

1. Responsabilidades Scrum Master.

**RTA**:

* Facilitar las ceremonias (Daily, sprint, review)
* Velar por el cumplimiento de Scrum.
* Eliminar los impedimentos para el equipo.
* Facilitador con el equipo de trabajo.

1. Definir complejidad ciclomática:

**RTA**:

* Es una métrica de SW cuantitativa para medir la complejidad lógica de un programa.
* Indica la cantidad de caminos independientes para testear las situaciones lógica de una pieza de código.

1. Lean: Se puede aplicar desde requerimientos hasta mantenimiento.

**RTA**:

* Verdadero. Es una filosofía de trabajo que no se limita a una etapa del ciclo de vida.

1. Lean: Está basado en principios de la mejora continua.

**RTA**:

* Verdadero. Es uno de los principios que predica la filosofía.

1. Lean: Uno de los principales objetivos es crear valor para el equipo de desarrollo.

**RTA**:

* Verdadero.

1. Lean: Involucra solo a colaboradores de IT.

**RTA**:

* Falso. Involucra a toda la organización y a los clientes.

1. Es un Framework para evaluar la madurez de los procesos de IT.

**RTA**:

* Falso. Es una filosofía de trabajo.

1. Testing: Prueba de caja negra para: “Solo podrán aplicar al beneficio extraordinario los nacidos en CABA entre los años 70 y 78 (ambos incluidos) y los nacidos en el año 83”. Declaraciones correctas.

**RTA**:

* Nacidos en 70 y 78 pertenecen a la misma clase de equivalencia.
* El nacido en el 78 y en el 83 tiene el mismo comportamiento.

1. SCM: Un IC puede no formar parte de una línea base.

**RTA**:

* Verdadero. Puede haber IC que no sean manejados en una línea base en particular.

1. SCM: Un release del producto “A” puede contener dos versiones del mismo IC.

**RTA**:

* Falso. Para un release solo puede haber una versión del mismo IC.

1. Una línea base puede no contener ningún IC que sea código.

**RTA**:

* Verdadero. Puede solo incluir documentación o modelos.

1. Scrum: “Juan es Scrum master. Maria es de su equipo y tiene preguntas sobre un requerimiento y el Producto Owner siempre está ocupado.” ¿Que acciones debe tomar Juan?

**RTA**:

* Organizar una reunión con el PO y explicarle las necesidades que tiene el equipo con el.

## **05/12/2022**

1. El diagrama de causa y efecto se usa para conocer los problemas del desarrollo de SW.

**RTA**:

* Falso. Se utiliza para conocer las causas relacionadas a un problema, independientemente del área en donde se aplique.

1. ¿Que es el SWEBOK y para que se utilizaría?

**RTA**:

* Es el Software Engineering Book of Knowledge e indica los requerimientos para calificar a un área de la ingenieria como profesional.
  + Tambien indica los conocimientos relativos para esa disciplina

1. Similitudes y diferencias entre Scrum y Kanban

**RTA**:

* Kanban es una forma de administración visual del flujo de trabajo.
  + Intenta realizar más con menos esfuerzo.
  + Kanban no posee roles o usuarios definidos para su uso.
  + Su principal distinción es que Limita el WIP, esto implica determinar la cantidad de tareas simultaneas que se pueden realizar en cada etapa del proyecto.
* Scrum es una metodología de trabajo para la administración de proyectos centrada en el trabajo en equipo.
  + Tiene sus roles definidos y obligatorios (Scrum master, Stakeholders, Equipo)
  + Define sus artefactos (Sprint y Product Backlog)
  + Define sus ceremonias (Daily, Retrospectiva, Sprint review)
* Similitudes:
  + Ambas sirven para proyectos incrementales.

1. Similitudes y diferencias entre Testing Exploratorio y Oráculos.

**RTA**:

* El testing exploratorio se basa en la experiencia y creatividad del tester para llevarse a cabo. Se le da la información mínima necesaria para que pueda realizar su trabajo y todo el proceso de diseño y creación de casos corre por su cuenta.
* Oráculos son principios o mecanismos que sirven para saber si un componente de SW está funcionando según lo establecido.
  + Para esto se utiliza la regla de HICCUPS.
* Similitudes:
  + Ambos se pueden utilizar para testing dinámico.
  + Ambos se puede utilizar para pruebas de caja blanca y negra.

1. Diferencias entre testeo estático y dinámico. Ejemplos de estático.

**RTA**:

* Testing dinámico necesita el SW funcionando para poder realizarse mientras que el estático no lo requiere.
* Ej: Cobertura de sentencias o complejidad ciclomática.

1. Métodos no paramétricos no estiman tamaño.

**RTA**:

* Falso. Todos estiman tamaño para, luego, poder estimar el esfuerzo.

1. SCM: Conviene tener más de un repositorio para tener más seguridad y productividad.

**RTA**:

* Falso. Conviene el uso de un único repositorio para poder conocer en todo momento donde estamos trabajando.
* En el caso de que se quiera aumentar la productividad, se pueden crear branchs, que son líneas base paralelas de desarrollo a partir de una misma configuración.
* En el caso que se quiera aumentar la seguridad se pueden crear nuevas baselines, que se definen como un conjunto de items de configuración que nos permiten tomar decisiones para determinar un acuerdo

1. Se tienen siempre los mismos ICs en una baseline pero pueden variar las versiones.

**RTA**:

* Falso. Pueden agregarse o sacarse ICs en cada baseline.

1. Tipos de pruebas dinámicas.

**RTA**:

* Test de stress, de volumen, de capacidad, de performance, Smoke Test.

1. MTBF y MTTR son buenas métricas para evaluar nuestros recursos.

**RTA**:

* Falso. Son métricas para evaluar la fiabilidad de nuestro sistema.
* MTBF es el tiempo promedio entre fallas que tiene un sistema.
* MTTR es el tiempo promedio que toma al sistema en recuperarse para poder ser usado nuevamente luego de la ocurrencia de una falla.

## **12/12/2022**

1. Un PM tiene un proyecto atrasado y considera sumar más personas para recuperar las demoras. ¿En que condiciones esto es válido?

**RTA**:

* Las tareas por delante son independientes, ya que no importa el estado de las demas, los nuevos van a poder trabajar.
* Las tareas requieren nula o baja comunicación entre ellas.
* Las tareas por delante requieren baja supervisión.

1. La visión del producto se fija en la adecuación al propósito.

**RTA**:

* Falso.
* La visión de usuario es la que se basa en la adecuación al propósito.
  + Es subjetiva.
* La visión del producto es la que se basa en características propias del producto.
  + Es objetiva.

1. Definir 5 atributos de calidad de SW y sus subatributos.

**RTA**:

* Adecuación funcional: Capacidad de un sistema para proporcionar funcionalidades que satisfagan las necesidades implícitas y explicitas del usuario.
  + Pertinencia funcional.
  + Completitud funcional.
  + Corrección funcional.
* Mantenibilidad: Capacidad de un sistema o componente para ser modificado de forma efectiva y eficiente debido a necesidades perfectivas, correctivas o evolutivas.
  + Modularidad.
  + Reusabilidad.
  + Capacidad para ser probado.
* Usabilidad: Capacidad de un sistema para ser entendido, aprendido, manejado y resultar atractivo al usuario.
  + Estética de la interfaz de usuario.
  + Operabilidad.
  + Accesibilidad.
* Fiabilidad: Capacidad de un sistema para desempeñar funciones específicas con una cantidad de recursos determinados durante un período de tiempo
  + Madurez
  + Disponibilidad.
  + Tolerancia a fallos.
  + Capacidad de recuperación.
* Compatibilidad: Capacidad de dos o más sistemas para intercambiar información y cumplir sus funciones compartiendo un mismo entorno de SW o HW.
  + Coexistencia.
  + Interoperabilidad.

1. Se arma una estrategia de prueba y se está considerando hacer caja negra o caja blanca. ¿Cual es incorrecta?

**RTA**:

* En caja blanca, si no se tienen los requerimientos escritos, se puede recurrir a heurísticas del tipo HICCUP.
  + HICCUP no es específico de caja blanca.
  + HICCUP no necesita los requerimientos escritos.

1. En el marco de entendimiento del contexto de un proyecto para tomar las mejores decisiones se identifica, según Cynefin, que se conoce perfectamente la relación causa-efecto. ¿En que contexto estas?

**RTA**:

* Simple. En este contexto se aplican las mejores prácticas, se conoce con certeza la relación entre una causa y su efecto y se elige siempre la mejor alternativa de entre todas las posibles.
  + Sense → Categorize → Respond

1. Ventajas y fortalezas de Planning Poker.

**RTA**:

* Como se estima el tamaño, un cambio en el equipo no modifica las tareas sino que modifica la velocidad del Sprint.
* Al utilizar sprints cortos, los desvíos se pueden corregir rápido.
* Se pierde poco tiempo en la fase de planificación.
* Se puede utilizar en etapas tempranas del proyecto.

1. Explicar el rol de Status & Accounting.

**RTA**:

* Es el encargado del seguimiento y control de los ICs que se encuentran gestionados en la configuración. Debe proporcionar información de administración efectiva, por ejemplo, los ICs aprobados, el estado de los cambios solicitados y un reporte de trazabilidad sobre la línea de base actual.

1. Dada una observación de auditoría, se tiene que replantear el proceso de Administración de cambios en producción. Indicar la incorrecta:

**RTA**:

* Los cambios de emergencia pueden impactarse y regularizarse luego.
  + Esto está mal porque un cambio primero debe ser evaluado y aprobado antes de pasar al entorno productivo, sea cual sea su grado de importancia, ya que sino puede ser una puerta hacia nuevos problemas.

# **Super Cuestionario**

## **1er Parcial:**

1. Estimaciones: Afirmaciones correctas.

**RTA**:

* Planning poker es una técnica de estimación basada en el consenso.
  + Es una técnica donde se estima, en un grupo, los tamaños de las tareas relativas a un pivote que se elige a priori.
* Object points es una técnica que se puede utilizar en proyectos de mantenimiento.
  + Es una técnica que se basa en identificar los componentes de un proyecto. A cada componente se le asigna un peso según su complejidad.
    - Su principal desventaja es que depende de como el proyecto esté construido.

1. ISO 25000: Definición del atributo Eficiencia de desempeño:

**RTA**:

* Es el desempeño relativo de un sistema a la cantidad de recursos utilizados bajo determinadas condiciones.

1. Afirmaciones sobre visión de manufactura.

**RTA**:

* Es la conformidad con las especificaciones.
* El marco CMMI utiliza los fundamentos de esta visión

1. Estimaciones: Afirmación correcta.

**RTA**:

* En Planning Poker, la opinión de los miembros influyentes del equipo es aislada del resto del grupo al menos durante la primera iteración.

1. CMMI: Afirmación correcta.

**RTA**:

* CMMI no es una metodología para el desarrollo de SW.
  + CMMI es un marco o modelo para la construcción de procesos. Es un modelo para la mejora continua y la evaluación de procesos de desarrollo, mantenimiento y operación de SW.

1. Timebox: Afirmación correcta.

**RTA**:

* Los ciclos de vida evolutivos son apropiados para el éxito de esta práctica.
  + Ya que se realiza en periodos cortos (de 1 a 3 meses).

1. Cynefin: Afirmación correcta en contexto “Complicado”.

**RTA**:

* Se aplican buenas prácticas solo por expertos.
  + Se necesita de un experto para poder discernir entre las alternativas y poder elegir la que mejor se adapte a la situación.

1. ISO 25000: La capacidad de protección contra acceso no autorizado a los datos se denomina:

**RTA**:

* Confidencialidad.

1. Kanban: Afirmaciones correctas.

**RTA**:

* Permite detectar cuellos de botella que producen demoras.
* Es útil aplicarlo en un escenario de resolución de incidencias.
* En el WIP, es el equipo el que decide cuanto trabajo comprometer.
* En todo momento tengo visualización del flujo de trabajo.
* Limitar el WIP me permite acordar la cantidad de cosas que se pueden realizar en paralelo.

1. Scrum: Afirmaciones incorrectas.

**RTA**:

* Scrum Master define las prioridades del proyecto.
* Product Owner es dueño del proceso.
* Scrum Master administra el ROI priorizando requerimientos.

1. ISO 25000: La métrica Complejidad ciclomática hace referencia al atributo:

**RTA**:

* Mantenibilidad.

1. Estimaciones: Para estimar un proyecto de forma consensuada donde no se tiene historia.

**RTA**:

* Planning Poker:
  + Ya que es un método no paramétrico y se realiza en grupo, haciendo una estimación de tamaño relativo a una historia pivote previamente seleccionada.

1. ISO 25000: La métrica MTTR es representativa del que atributo:

**RTA**:

* Fiabilidad.
  + Indica el tiempo promedio de recuperación del sistema luego de un fallo.

1. Timebox: Afirmación incorrecta.

**RTA**:

* Solo puede modificarse el deadline para incorporar requerimientos.

1. ISO 25000: La capacidad de un sistema o componente de prevenir accesos o modificaciones no autorizados es:

**RTA**:

* Integridad.

1. ISO 25000: Métricas de mantenibilidad.

**RTA**:

* Complejidad ciclomática.
* Cantidad de métodos por clase (Reusabilidad o LoC)

1. Scrum: Responsabilidad del Scrum Master.

**RTA**:

* Quitar obstáculos del camino.
* Proteger y cuidar el equipo.
* Facilitador con el equipo de trabajo.
* Dueño del proceso.
* Debe estar pendiente del cumplimiento del proceso.

1. ISO 25000: Está alineado con la visión de calidad.

**RTA**:

* De producto.

1. Kanban: Afirmaciones incorrectas.

**RTA**:

* Una vez comenzado el trabajo no se admiten cambios.

1. Cynefin: Contexto complicado:

**RTA**:

* Detectar, analizar y responder (Sense → Analyze → Respond)

1. Estimaciones: Afirmaciones correctas.

**RTA**:

* FP permite medir el tamaño del SW en base a la funcionalidad definida en los requerimientos.

1. Lean: Afirmación incorrecta.

**RTA**:

* Es un framework para evaluar la madurez de los procesos de IT.
  + Lean es una filosofía de trabajo y un enfoque para la mejora continua.
  + Se basa en eliminar los residuos y todo el NO valor agregado.
  + Su objetivo es ofrecer una mayor calidad y un decremento en los costos y tiempos del ciclo.

1. ISO 25000: Métricas mantenibilidad.

**RTA**:

* Complejidad ciclomática.
* LoC.

1. Calidad: Visiones objetivas.

**RTA**:

* De manufactura: Implica la conformidad con las especificaciones.
  + Se establece un proceso para las buenas prácticas y generar un acuerdo sobre lo que se define como calidad.
  + Se definen parámetros iguales para todos.
* Del producto: Vinculado a características propias del producto.
  + Esta asociado a la ISO 25000.
  + Se evalua según ciertos parámetros.

1. Proyecto: Afirmaciones correctas:

**RTA**:

* El grado de libertad es aquella dimensión que tiene la mayor flexibilidad.
  + La restricción es un requerimiento del cual no tenemos control y tiene nula flexibilidad.

1. Timebox: En un proyecto de 120 días con un equipo entrenado en la tecnología y con requerimientos priorizados, deciden aplicar Timebox. ¿En que momento lo aplicaría?

**RTA**:

* Etapa de desarrollo.

## **2do Parcial:**

1. Métricas de mantenibilidad:

**RTA**:

* Complejidad ciclomática.
* LoC

1. Afirmación falsa sobre SCM.

**RTA:**

* Para tener un Deploy Continuo no se necesita previamente un proceso de Delivery Continuo.
  + Delivery continuo es la automatización de los deploys hasta el entorno pre productivo inclusive.
  + Deploy continuo es lo mismo pero teniendo deploys automáticos hasta el entorno productivo. Es más complejo de implementar.

1. Proceso de SCM para identificar todos los cambios realizados por un Dev en los últimos 5 meses.

**RTA**:

* Status & Accounting. Son los encargados del control y la gestión de los ICs que se encuentran manejados por la configuración.
  + Se deben generar reportes para lograr una administración efectiva:
    - Listado con los ICs aprobados.
    - Estado de todos los cambios.

1. Integración continua: Cual de los siguientes es **incorrecto**.

**RTA**:

* El build debe ser aprobado por el Scrum Master [Proceso automático]
  + Integración continua no depende de utilizar o no Scrum.
  + Al utilizar Scrum como metodología, la aprobación de los cambios depende del PO.

1. Exploratory Testing. Marcar la incorrecta:

**RTA**:

* El trabajo creativo para la prueba se hace antes de la ejecución, durante el armado del charter.
  + La totalidad de ET es un proceso creativo, ya que se van descubriendo nuevos casos durante la ejecución del mismo.
  + El charter funciona como una guía pero no limita lo que el tester puede hacer.

1. Auditoria: Se descubre un fraude producto de un “Backdoor” en el código. ¿En que auditoría se debería haber descubierto?

**RTA**:

* Física. Es la encargada de verificar la estructura especifica, así como también de validar que el código subido coincide con el cambio y, a su vez, que no fue subido nada que realice funciones no solicitadas.

1. Testing: Para mejorar los resultados del equipo. Indicar la incorrecta.

**RTA**:

* A un menor Índice de severidad por defectos abiertos implicará una mejor calidad de desarrollo [Mejor release readiness]

1. Cobertura de código: Afirmación incorrecta.

**RTA**:

* La complejidad ciclomática es inversamente proporcional al esfuerzo de mantenibilidad del código.
  + Es directamente proporcional.

1. Testing: Se considera hacer caja blanca y caja negra. Marcas las incorrectas:

**RTA**:

* En caja blanca, si no se tienen los requerimientos escritos, se puede usar Hiccups heuristic.
* A mayor cohesión de los componentes de SW, más compleja es la prueba.
  + A mayor cohesión, se simplifican las pruebas porque cada módulo tiene una función bien determinada dentro del sistema.

1. Release del producto “A”. Indicar incorrecta.

**RTA**:

* En esta release hay un mismo IC con dos versiones distintas.
  + Para cada release, solo puede haber una versión del mismo IC.
  + No obstante, en distintas releases puede haber distintas versiones del mismo IC, la misma versión o directamente no estar.

1. Clasificando los IC que gestionará en el proyecto. Marcar incorrecta:

**RTA**:

* El diseño de arquitectura es un IC del proyecto.

1. Testing: Situación: Se aceptan depósitos de las sucursales comprendidas entre la 50 y la 99 y además la 115. Se identificaron 2 clases validas y tres invalidas. Indicar afirmación correcta.

**RTA**:

* [50 - 99] U {115}
* Se necesitaran como mínimo 3 casos para cubrir los casos de borde (son 7) {49,50,99,100,114,115,116}

1. Testing: Pruebas no funcionales: Seleccionar la correcta.

**RTA**:

* Ninguna opción que se haya definido en la cuantificación de requerimientos.

1. Testing: Seleccionar correcta.

**RTA**:

* La prueba de aceptación de usuario es realizada para verificar que el sistema se ajusta a los requerimientos y la ejecutan los mismos usuarios.

1. SCM: Afirmaciones correctas.

**RTA**:

* Durante un proyecto se pueden establecer tantos baselines como se quiera.
* Una nueva versión de un producto y un nuevo baseline pueden coincidir.

1. Exploratory Testing: Afirmaciones correctas

**RTA**:

* Se puede utilizar para atacar un riesgo en particular.
* Depende del conocimiento que se obtiene durante la ejecución.

1. SCCB: Afirmaciones correctas.

**RTA**:

* SCCB gobierna el proceso de control de cambios.
* SCCB tiene autoridad para rechazar un cambio.

1. SCM: Problemas que puede resolver Status & Accounting.

**RTA**:

* Falta el diagrama de clases en la vaselina de diseño.
* Conocer el alcance de un cambio en determinado componente.

1. Testing: Situación: Solo pueden aplicar al nuevo crédito bancario aquellas PYMES que tengan cuenta en el banco y no tengan deuda en CC en los últimos 15 meses.

**RTA**:

* Valida: Con Cuenta y Deuda < 15 meses.
* Invalido: Sin cuenta + Sin cuenta y N meses + Con cuenta y >= 15 meses.
* Con cuenta y 15 meses + Con cuenta y 16 meses pertenecen a la misma clase de equivalencia.

1. Testing: Indicar correcta.

**RTA**:

* Una partición son todos los posibles casos de prueba que se dividen en clases.

1. Testing: Indicar correcta.

RTA:

* Identificación de clases de equivalencia se hace dividiendo cada condición de entrada en clase valida e invalida.
* En caja blanca, el grado de cobertura de condición es más completa que la cobertura de decisión.
* Dos fallas distintas que son detectadas pueden ser ocasionadas por el mismo defecto.