**PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL**

**FORMATO GUÍA DE APRENDIZAJE**

1. **IDENTIFICACIÓN DE LA GUIA DE APRENIZAJE:** GUÍA **REQUERIMIENTOS - SCRUM**

* Denominación del Programa de Formación: **TECNÓLOGO EN ANÁLISIS DESARROLLO DE SISTEMAS DE INFORMACIÓN**
* Código del Programa de Formación: **228106**
* Nombre del Proyecto: **SISTEMA INTEGRAL WEB PARA GESTION DE PROCESOS EDUCATIVOS DEL CEET**
* Fase del Proyecto: **Planeación.**
* Actividad de Proyecto: **Sistema Integral Web Para Gestión De Procesos Educativos Del Ceet.**
* Competencia: **220501032 - Analizar los requisitos del cliente para construir el sistema de información**
* Resultados de Aprendizaje Alcanzar: **Interpretar el informe de requerimientos, para determinar las necesidades tecnológicas en el manejo de la información, de acuerdo con las normas y protocolos establecidos en la empresa.**
* Duración de la Guía: **22 HORAS**

1. **PRESENTACIÓN**

**Scrum** es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

1. **FORMULACION DE LAS ACTIVIDADES DE APRENDIZAJE**

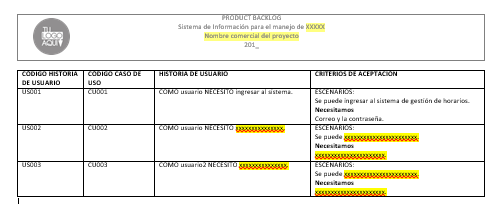


* 1. **- Material de apoyo y conceptos:** Para iniciar con las actividades de la presente guía realizaremos un vistazo al video **RUP y SCRUM** el cual proporcionara los conocimientos necesarios para diferenciar las características de **2 de las metodologías de desarrollo** más usadas en la creación de Software. **TIEMPO – 2 HORAS (Grupal).**
  2. **– Cuadro comparativo:** según el video relacionado anteriormente realice un cuadro comparativo de las dos **metodologías de desarrollo** **RUP** y **SCRUM**. **TIEMPO – 2 HORAS (Individual).**

|  |  |  |
| --- | --- | --- |
| **CUADRO COMPARATIVO** | | |
| **CRITERIO** | **RUP** | **SCRUM** |
| **CONCEPTO** | Es un proceso que suministra un enfoque para asignar tareas y responsabilidades de una organización desarrollado. | Conjunto de buenas prácticas para trabajar colaborativamente, en equipo y obtener un mejor resultado posible en un proyecto es decir es un ciclo completo. |
| **VENTAJAS** | * Mayor documentación. * Configuración y control de cambios. * Es modelado guiado por caso de uso. * Es cerrado en arquitectura por riesgo. * Verifica la calidad de software. | * Conocimiento necesario para lograr un objetivo. * Involucra desde un principio y se da un rol a todos. * Entregables en tiempo y forma. |
| **DESVENTAJAS** | * Los cambios son en una fase. * Son de proyecto grandes. | * Los miembros del equipo se saltan pasos importantes en el camino para llegar el Sprint final. * Demasiadas reuniones para poco avance. |
| **ROLES** | * Analistas. * Desarrolladores. * Gestores * Apoyo y especialistas * Coordinación de revisiones. | * Trabajo realizado por el equipo de especialistas. * Visibles, transparente por el equipo de especialistas. * Responsabilidad. |
| **CARACTERISTICAS** | * Se abarca prácticas de gestión sin entrar en prácticas de desarrollo. * Delega completamente en el equipo la responsabilidad. | * Fundamenta en principios. * Reduce el costo del cambio en todas las etapas. * Equipo con formación elevada. |

|  |  |  |
| --- | --- | --- |
| **CRITERIOS** | **RUP** | **SCRUM** |
| **TIPO DE FRAMEWORK** | Análisis diseño implementación y documentación de sistemas orientados a objetos. | Gestión de desarrollo de un software basado en un proceso iterativo e incremental. |
| **TIPO DE REVISION** | En cada fase se realiza una o mas iteraciones, perfeccionando así los objetivos. Sino se termina fase continua con la otra. | Sus pruebas se realizan al finalizar el proceso de enfatizando en la reutilización de los componentes de los programas ya comprobados. |
| **TIPO DE DESARROLLO** | Proceso iterativo  Incremental por fases:  INICIO  ELABORACION  CONSTRUCCION  TRANSMISION | Desarrollo simple que requiere trabajo duro.   * Control de forma empírica y adaptable a la evaluación del proyecto. |
| **FACILIDAD DE USO** | Dirigido por casos de uso:   * Establecimiento temprano de una buena arquitectura. * Iterativo e incremental. * Incrementa el trabajo se divide en mini proyectos. | * No se basa en el seguimiento de un plan. * Modelo adaptable. * Construcción incremental basado en iteraciones. * No existe trabajo con diseño o abstracción. |

|  |  |  |
| --- | --- | --- |
| **NOMBRE** | **RUP** | **SCRUM** |
| **ENFOQUE** | Iterativo | Iterativo |
| **CICLO** | Ciclo formal se define a través de 4 fases, pero algunos flujos de trabajo pueden ser concurrentes. | Cada Sprint (Iteración) es un ciclo completo. |
| **PLANIFICACION** | Plan de proyecto formal, asociada a múltiples iteraciones, se utiliza. El plan es impulsado fecha final y también cuenta objetivo intermedio. | No de extremo a extremo del plan de proyecto. Cada plan de la siguiente iteración se determina al final de la iteración actual (No la fecha final de tracción). Dueño del producto (Usuario de negocio clave) determina el momento en que el proyecto se lleva cabo. |
| **ALCANCE** | Ámbito de aplicación esta predefinido antes del inicio del proyecto y se documenta en el documento de alcance. Ámbito de aplicación pueden ser revisados durante el proyecto, los requisitos se están aclarando, pero estas modificaciones están sujetas a un procedimiento estrictamente controlado. | En vez de alcance, SCRUM utiliza una cartera de proyecto, que se reevaluado al final de cada Iteración (sprint). |
| **LOS ARTEFACTOS** | Visión/Ámbito de aplicación del documento, el paquete formal de requisitos funcionales, documento de arquitectura del sistema, plan de desarrollo, plan de pruebas, scripts de pruebas, etc.… | EL único artefacto formal es el software operativo. |
| **TIPO DE PROYECTO/PRODUCTO** | Recomendado para grandes, a largo plazo, a nivel de empresa con proyecto a medio y alta complejidad. | Recomendado para las mejoras rápidas y organizaciones que no dependen de una fecha limite. |

* 1. **Historias de Usuario:** Como se trabajó en la anterior guía, la empresa **Peluquería - XYZ** desea implementar su sistema de información y actualizarlo, por ello y basándose en los **Requerimientos Funcionales y NO funcionales** que usted identifico anteriormente, se implementara la metodología **SCRUM** y se creara las **Historias de usuario** que permitirán describir más claramente las necesidades de cada uno de los usuarios del sistema, los cuales según la peluquería son el **administrador** y los **clientes**. **TIEMPO – 2 HORAS (Individual).**
  2. **Historias de Usuario Proyectos:** Crear las **Historias de usuario** correspondientes a sus proyectos de formación **Individual** y **Grupal**. **TIEMPO – 6 HORAS (Individual).**



* 1. **– Introducción SCRUM:** Continuando con las temáticas y reforzando los conocimientos realice la siguiente investigación de términos necesarios para el manejo de SCRUM, con ayuda de video suministrado por el instructor y lo investigado en Formación. **TIEMPO – 2 HORAS (Individual).**

1. ¿Qué es Scrum?
2. ¿Cuáles son los Roles en Scrum?
3. ¿Qué son los **Issues** y de ejemplos?
4. ¿Qué es **Pipeline**?
5. ¿Qué se hace en Pipeline **Backlog**?
6. ¿Qué se hace en Pipeline **In Progress**?
7. ¿Qué se hace en Pipeline In **Review/QA**?
8. ¿Qué es **Milestones**?
9. ¿Cómo se llama el reporte más usado en **Scrum**?
10. ¿Qué son **Boards** y cuáles son sus componentes básicos?
11. ¿Qué son y cuál es el funcionamiento de **Assignees**?
12. ¿Qué son los **Epics** y de ejemplos?
13. ¿Qué son los **estimates** y cuáles son los valores?

**SOLUCION**

1. Scrum es una forma o marco de trabajo para un desarrollo ágil de software, también es un proceso en el que se aplican de manera regular un conjunto de buenas practicas para trabajar colaborativamente, en equipo y obtener un mejor resultado posible de proyecto.

## Los Roles en Scrum son:

## Roles Principales:

* **Product Owner**

El *Product Owner* se asegura de que el equipo Scrum trabaje de forma adecuada desde la perspectiva del negocio. El Product Owner ayuda al usuario a escribir las historias de usuario, las prioriza, y las coloca en el Product Backlog.

* **ScrumMaster (o Facilitador):** El *Scrum* es facilitado por un *ScrumMaster*, cuyo trabajo primario es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El *ScrumMaster* no es el líder del equipo (porque ellos se auto-organizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga. El ScrumMaster se asegura de que el proceso Scrum se utiliza como es debido. El ScrumMaster es el que hace que las reglas se cumplan.
* **Equipo de desarrollo:** El equipo tiene la responsabilidad de entregar el producto. Es recomendable un pequeño equipo de 3 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc).

### Roles Auxiliares

Los roles auxiliares en los "equipos Scrums" son aquellos que no tienen un rol formal y no se involucran frecuentemente en el "proceso Scrum", sin embargo, deben ser tomados en cuenta. Un aspecto importante de una aproximación ágil es la práctica de involucrar en el proceso a los usuarios, expertos del negocio y otros interesados ("[stakeholders](https://es.wikipedia.org/wiki/Stakeholder)"). Es importante que esa gente participe y entregue retroalimentación con respecto a la salida del proceso a fin de revisar y planear cada sprint.

* Stakeholders (Clientes, Proveedores, Vendedores, etc.)

Son las personas que hacen posible el proyecto y para quienes el proyecto producirá el beneficio acordado que justifica su desarrollo. Solo participan directamente durante las revisiones del "sprint".

1. Se atribuye a la unidad de trabajo para realizar una mejora en un Sistema informático.

**EJEMPLO:**

Un issue puede ser el arreglo de un fallo, una característica pedida, una tarea, un pedido de documentación específico y todo tipo de solicitud al equipo de desarrollo.

1. En programación. La **arquitectura en pipeline** (basada en filtros) consiste en ir transformando un flujo de datos en un proceso comprendido por varias fases secuenciales, siendo la entrada de cada una la salida de la anterior.

En informática, una **tubería** (*pipeline* o cauce) consiste en una cadena de procesos conectados de forma tal que la salida de cada elemento de la cadena es la entrada del próximo. Permiten la comunicación y sincronización entre procesos. Es común el uso de búfer de datos entre elementos consecutivos.

La comunicación por medio de tuberías se basa en la interacción productor/consumidor, los procesos productores (aquellos que envían datos) se comunican con los procesos consumidores (que reciben datos) siguiendo un orden FIFO. Una vez que el proceso consumidor recibe un dato, este se elimina de la tubería.

1. El Backlog ayuda a regular el flujo de trabajo y brinda a la gerencia mejor capacidad para asegurar que el personal del Portafolio Activo está ocupado, pero no desbordado. Se termina más trabajo y el trabajo que se termina es el de mayor prioridad.
2. El pipeline in progress es el proceso que se va llevar a cabo en los días realizado en un proyecto por lo tanto es el progreso de cada dia en un proyecto.
3. Una tubería de control de calidad rápida y confiable generalmente se basa en pruebas de automatización de varios niveles y una integración continua estable, que ejecuta todas estas pruebas a diario en diferentes entornos de prueba. No hace mucho tiempo, configurar, configurar y mantener un flujo de trabajo completo, era un gran desafío. Pero las innovaciones que se están produciendo en las metodologías de control de calidad y las aplicaciones de infraestructura ahora le permiten dedicar un tiempo mínimo a este frente, para que pueda concentrarse más en su aplicación.
4. Un hito es un punto de referencia que marca un evento importante de un proyecto y se usa para supervisar el progreso del proyecto. Todas las tareas que tengan una duración cero se muestran automáticamente como un hito.
5. El reporte más usado en Scrum son los Sprint.
6. Es el dispositivo que funciona como la plataforma o circuito principal de una computadora, integra y coordina todos los sus demás elementos. Tambien es conocida como placa base, placa central, placa madre, tarjeta madre o **Board** (en inglés motherboard, mainboard).

**Los componentes que contiene son:**

* Microprocesador y Zócalo (Socket) del microprocesador.
* Memorias y ranuras de memoria.
* La BIOS.
* Ranuras de expansión.
* Conectores internos y conectores eléctricos.
* Conectores externos y elementos integrados variados.
* Chipset de control.

1. La definición de cesionario (**Assignees)** en el diccionario es una persona a quien se transfiere algún derecho, interés o propiedad. Otra definición de cesionario es un convicto que ha sido asignado.
2. Este paradigma también se conoce como arquitecturas de *Independencia*. Fue utilizado por Intel y HP para el desarrollo de la arquitectura de Intel IA-64 y se ha implementado en la línea de procesadores de servidor Intel Itanium e Itanium 2. El objetivo de EPIC era aumentar la capacidad de los microprocesadores para ejecutar instrucciones de software en paralelo mediante el uso del compilador, en lugar de la compleja circuitería en cápsula (die), para identificar y aprovechar las oportunidades para la ejecución en paralelo. Esto permitiría escalar el rendimiento más rápidamente en los futuros diseños de procesadores, sin tener que recurrir a frecuencias de reloj cada vez más altas, las cuales se han convertido desde ese momento en una

**Ejemplo 1:** Reportes de desempeño de ventas  
  
Aquí la épica:  
  
Como Vicepresidente de mercadeo y ventas, quiero revisar el desempeño histórico de las ventas, para poder identificar las regiones geográficas y productos de mejor desempeño  
  
Esta épica se puede subdividir en:  
  
Como VP de Mercadeo, quiero seleccionar el período de tiempo en el cual realizaré la revisión de las ventas.  
  
Como VP de Mercadeo, puedo clasificar la información de ventas por región geográfica y productos.  
  
**Ejemplo 2:**Maximizar los ingresos de un hotel  
  
Como un operador hotelero, quiero establecer las tarifas óptimas para las habitaciones de mi hotel.  
  
Esta historia se puede subdividir en:  
  
Como un operador hotelero, quiero establecer la tarifa óptima para las habitaciones en base a los precios del año anterior.  
  
Como un operador hotelero, quiero establecer la tarifa óptima para las habitaciones en base a las tarifas de otros hoteles comparables con el mío. problemática importante debido a problemas de alimentación y refrigeración.

1. La estimación es el proceso de encontrar una estimación, o aproximación, que es un valor que puede utilizarse para algún propósito, incluso si los datos de entrada pueden ser incompletos, inciertos o inestables. No obstante, el valor es utilizable porque se deriva de la mejor información disponible.
   1. **– Mapa conceptual:** Antes de implementar **SRCUM** en alguna herramienta de desarrollo se realizará un mapa conceptual para reforzar esas terminologías vistas en el video y en la investigación previa. **TIEMPO – 2 HORAS (Individual).**



* 1. **– Manejo de ZenHub:** En esta actividad se relacionará cada una de las historias de usuario identificadas en el punto 3.4 correspondientes a su proyecto Individual, la herramienta a usar es ZenHub, la cual ayudara a poner en práctica todo lo visto en el manejo de Scrum en el transcurso de esta Guía. **TIEMPO – 2 HORAS (Individual).**

* 1. **– Implementación Proyecto:** Asociar las correspondientes **ISSUES** obtenidas en sus **Proyectos Grupales** en la herramienta vista previamente **ZenHub**. **TIEMPO – 2 HORAS (Grupal).**



* 1. **– Pictionary Requerimientos:** Concluyendo el desarrollo de la guía, se implementará la actividad en **Pictionary** que permitirá recopilar conceptos y terminología de la presente guía – **Requerimientos** y **Scrum**. **TIEMPO – 2 HORAS (Grupal).**

1. **ACTIVIDADES DE EVALUACIÓN**

|  |  |  |
| --- | --- | --- |
| **Evidencias de Aprendizaje** | **Criterios de Evaluación** | **Técnicas e Instrumentos de Evaluación** |
| **Evidencias de Conocimiento :**  **Evidencias de Desempeño:**  **Evidencias de Producto:** | Elabora propuestas de trabajo, de acuerdo con la interpretación de las necesidades tecnológicas, expuestas en el informe de requerimientos, según normas y protocolos de la empresa. | 1. Cuadro Comparativo 2. Investigación 3. Mapa Conceptual 4. Historias de Usuario 5. Manejo ZenHub |

**5. GLOSARIO DE TERMINOS**

* **Los requerimientos funcionales:** definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas.
* **Los requerimientos no funcionales:** tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc.

**6. REFERENTES BIBLIOGRÁFICOS**

* Bourque, P., Fairley, R. (2014). Guide to the Software Engineering Body of Knowledge. IEEE Computer Society.
* PMI Project Management Institute (2013). A guide to the Project Management Body of Knowledge PMBOK Guide 5th Edition. Pennsylvania: PMBOK

**7. CONTROL DEL DOCUMENTO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Nombre** | **Cargo** | **Dependencia** | **Fecha** |
| **Autor (es)** | Javier Leonardo Pineda Uribe | Instructor | Contratista | 14 de Marzo de 2020 |

**8. CONTROL DE CAMBIOS** (diligenciar únicamente si realiza ajustes a la guía)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Nombre** | **Cargo** | **Dependencia** | **Fecha** | **Razón del Cambio** |
| **Autor (es)** |  |  |  |  |  |