**¿QUÉ ES LA INGENIERÍA DEL SOFTWARE?**

Aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software, es decir, la aplicación de la ingeniería al software.

¿QUÉ ES UNA PROFESIÓN RECONOCIDA?

Para que la ingeniería del software sea una legítima disciplina y reconocida profesión, es imperativo un consenso sobre el cuerpo de conocimiento. Este hecho es bien ilustrado por Starr cuando define que puede considerarse como legítima disciplina reconocida profesión. En su libro, ganador del premio Pulitzer, sobre la historia de la profesión médica de los EE.UU, indica:

La legitimización de una autoridad profesional envuelve tres afirmaciones distintivas: primero, todo conocimiento y competencias de un profesional han sido validadas por una comunidad de compañeros de profesión; segundo, el conocimiento validado consensuadamente se basa en fundamentos científicos racionales; y tercero, las opiniones profesionales y consejos están orientadas hacia valores fundamentales como la salud. Estos aspectos de legitimidad corresponden con los tipos de atributos generalmente cubiertos en el término profesión

¿CUÁLES SON LAS CARACTERÍSTICAS DE UNA PROFESIÓN?

* Una educación profesional inicial en un currículum validado por una sociedad de acreditación.
* Registro de la correcta práctica por medio de una certificación voluntaria o licencia obligatoria.
* Habilidad espacial de desarrollo y una continua educación profesional.
* Soporte comunitario por medio de una sociedad profesional.
* Un compromiso con las normas de conducta a menudo prescritas en un código de ética.

¿CUÁLES SON LOS OBJETIVOS DEL PROYECTO SWEBOK?

La guía al cuerpo de conocimiento de ingeniería del software se estableció con los siguientes 5 objetivos:

1. Promover una visión consistente de la ingeniería del software en el mundo.
2. Clarificar la situación y definir fronteras de la ingeniería del software con respecto a otras disciplinas como la informática, gestión de proyectos, ingeniería informática y matemáticas.
3. caracterizar los contenidos de la disciplina de la ingeniería del software
4. Proporcionar al cuerpo de conocimiento de la ingeniería del software con los temas de interés
5. Proporcionar una base para el desarrollo planes de estudio, certificaciones individuales y materiales para licencias.

ORGANIZACIÓN JERÁRQUICA

La guía utiliza una organización jerárquica para descomponer cada área de conocimiento en un conjunto de temas catalogados. Una descomposición en 2 o 3 niveles proporciona una manera razonable de encontrar los temas. La guía trata los temas seleccionados de una manera compatible con las escuelas de pensamiento mayoritarias y con las descomposiciones encontradas en las organizaciones, literatura y estándares. La descomposición no presupone ningún dominio de aplicación particular, forma de negocio, filosofía de gestión, métodos de desarrollo, etc.

MATERIALES DE REFERENCIA Y MATRICES

Para proporcionar el acceso a los temas de interés de la guía el cuarto de los objetivos del proyecto la guía identifica material de referencia para cada AC, incluyendo capítulos de libro, artículos u otras fuentes de reconocido prestigio. Cada AC incluye una matriz relacionando la literatura con los temas. El total de la literatura citada intenta ser el adecuado para un graduado con 4 años de experiencia.

PROFUNDIDAD DEL TRATAMIENTO

Una de las cuestiones que surgieron desde el comienzo del proyecto, fue el nivel de detalle que la Guía debería proporcionar.

El equipo del proyecto adaptó un enfoque que ayuda con el quinto de los requerimientos proveer la base para el desarrollo curricular, certificaciones y licencias. El equipo editorial aplico el criterio de conocimiento generalmente aceptado, conocimiento consensuado, distinguiéndolo de conocimientos avanzados o de investigación (en base a la madurez) y de conocimiento especializado (en base a la generalidad de aplicación). La definición viene del Project Management Institute (PMI)

El conocimiento generalmente aceptado se aplica a la mayoría de los proyectos la mayoría del tiempo, y su amplio consenso valida su valor y efectividad

Sin embargo, el término generalmente aceptado, no implica que el conocimiento nombrado deba aplicarse uniformemente a todos los proyectos software cada proyecto necesita determinarlos, pero implica que todos los ingenieros competentes deberían de estar equipados con estos conocimientos. Siendo más precisos, el conocimiento generalmente aceptado debería ser incluido en los materiales de estudio de licencias que los graduados deberían de adquirir tras cuatro años de experiencia laboral

LIMITACIONES RELACIONADAS CON EL FORMATO DEL LIBRO

El formato en el cual este libro ha sido concebido tiene sus limitaciones. La naturaleza de los contenidos podría mostrarse mejor en formato de hipertexto, donde cada tema podría ser unido a otros temas que no sean el anterior y posterior a una lista.

Los enlaces entre las AC, no son del tipo entrada-salida. Las áreas de conocimiento proporcionan vistas al conocimiento que uno debería poseer con respecto a cada AC en la ingeniería del software. La descomposición de la disciplina dentro de cada AC y el orden en el cual las áreas de conocimiento son presentadas, no tienen por qué integrarse con ningún método o modelo particular. Los métodos son descritos en las apropiadas AC dentro de la guía, y la guía en sí misma no es parte de ellos.

REQUERIMIENTOS DEL SOFTWARE

Un requerimiento se define como una propiedad que debe exhibir el software para resolver algún problema del mundo real.

La primera subárea de conocimiento es Fundamentos de los Requerimientos del Software. Incluye las definiciones de requerimientos del software y sus principales tipos: producto vs. proceso, funcional vs. no funcional, propiedades emergentes. La subárea además describe la importancia de que los requerimientos cuantificables y distingue entre sistemas y requerimientos software.

La segunda subárea de conocimiento son los requerimientos del proceso, el cual introduce el proceso y orienta las 5 subáreas restantes y cómo la ingeniería de requerimientos encaja en otras con otros procesos de la ingeniería del software. Describe los modelos de proceso, actores del proceso, procesos de soporte y gestión, y la calidad mejora del proceso

La tercera subárea, es la Captura de requisitos, la cual se centra en de dónde vienen los requerimientos y cómo el ingeniero de software puede obtenerlos. Incluye las fuentes de los requerimientos y las técnicas de captura.

La cuarta subárea, Análisis de requerimientos, se centra en los procesos de análisis de requerimientos

DISEÑO DEL SOFTWARE

Según la el diseño es el proceso de definir la arquitectura, componentes, interfaces y otras características de un sistema o componente y el resultado de [ese] proceso. El AC está dividido en 6 subáreas.

-Fundamentos del Diseño del Software

-Temas Clave en el Diseño del Software

-Estructura del Software y la Arquitectura

- Calidad Evaluación de las del Diseño del Software

-Notaciones del Diseño del Software

-Estrategias y Métodos del diseño del Software

CONSTRUCCIÓN DEL SOFTWARE

La construcción del software se refiere la creación de software mediante una combinación de codificación, verificación, pruebas unitarias, pruebas de integración, y depuración. Está dividida en tres subáreas.

La primera subárea son los Fundamentos de la Construcción del Software. Los tres primeros temas son los principios básicos de la construcción: minimización de la complejidad, anticipación al cambio, y la construcción y verificación.

La segunda subárea describe la Gestión de la Construcción. Los temas son la construcción de modelos, planificación de la construcción y la medición de la construcción.

La tercera subárea cubre las Consideraciones Practicas. Los temas son el diseño de la construcción, lenguajes de construcción, codificación, pruebas de construcción.

PRUEBAS DEL SOFTWARE

Las pruebas de software se componen de la verificación dinámica del comportamiento de un programa con un conjunto finito de casos de pruebas, adecuadamente seleccionados del un infinito número de posibles ejecuciones del dominio.

MANTENIMIENTO DEL SOFTWARE

Una vez en producción, se descubren anomalías, los entornos de trabajo cambian y aparecen nuevos requerimientos de trabajo. La fase del ciclo de vida mantenimiento comienza una vez entregado el sistema, sin embargo, las actividades de mantenimiento ocurren mucho antes. El AC de mantenimiento del software está dividido en 4 subáreas.

-Fundamentos del Mantenimiento del Software

-Clave del Mantenimiento del Software

-Proceso de Mantenimiento

GESTIÓN DE LA CONFIGURACIÓN DEL SOFTWARE

La Gestión de la Configuración del Software (GCS) es la disciplina de la identificación del software en distintos puntos en el tiempo con el propósito de controlar los cambios sistemáticamente, y del mantenimiento de la integridad y trazabilidad de la configuración durante todo el ciclo de vida. Esta AC incluye 6 subáreas.

GESTIÓN DE LA INGENIERÍA DEL SOFTWARE

El AC de de la Gestión de la Ingeniería del Software trata la gestión y medición de la ingeniería del software. Mientras la medición es aspecto importante en todas las AC, es aquí donde se presenta el tema de programas de medición. Hay seis subáreas en la gestión de ingeniería del software. Las 5 primeras cubren la gestión de proyectos software y la última describe los programas de medición software

- Iniciación y Definición del Alcance

-Planificación de Proyectos Software

-Promulgación del Proyecto Software

-Revisión y Evolución

PROCESO DE LA INGENIERÍA DEL SOFTWARE

El AC de proceso de la ingeniería del software se centra en la definición, implementación, evaluación, gestión, cambio y mejora del proceso de ingeniería del software. Está dividido en cuatro subáreas.

-Proceso de Implementación y Cambios

-Definición del Procesos

-Evolución del Proceso

-Mediciones de Proceso y de Producto

HERRAMIENTAS Y MÉTODOS EN LA INGENIERÍA DEL SOFTWARE

El AC de Herramientas y Métodos de la ingeniería del software incluye ambas, herramientas de la ingeniería del software y métodos de la ingeniería del software.

Las subárea de Herramientas de la Ingeniería del Software utiliza la misma estructura que la guía en sí misma, con un tema por cada una de las otras nueve AC de la ingeniería del software. Se añade un tema adicional: cuestiones varias sobre herramientas como técnicas de integración de herramientas que son potencialmente aplicables a todo tipo de herramientas.

La subárea de Métodos de Ingeniería del Software se divide en cuatro subsecciones: métodos heurísticos que tratan aproximaciones informales, métodos formales basados en aproximaciones matemáticas, métodos de prototipado tratando varias formas de prototipados.

CALIDAD DEL SOFTWARE

El AC de la calidad del software se ocupa de las consideraciones sobre la calidad del software las cuales transcienden los procesos del ciclo de vida del software. Al ser la calidad del software un tema ubicuo a toda la ingeniería del software, también es considerada en muchas otras AC, por lo que el lector notará referencias a otras AC en toda esta AC.

DISCIPLINAS RELACIONADAS DE LA INGENIERÍA DEL SOFTWARE

El último capítulo se titula Disciplinas relacionadas de la Ingeniería del Software. Para circunscribir la ingeniería del software, es necesario identificar las disciplinas con las que la ingeniería del software comparte una frontera común.

**Requisito del software**

Básicamente, un requisito del software es una característica que se debe exhibir para solucionar un cierto problema en el del mundo real. La guía se refiere a requisitos de software‖ porque se refiere a los problemas que se tratarán por el software. Por lo tanto, un requisito del software es una característica que se debe exhibir por el software desarrollado o adaptado para solucionar un problema particular. El problema puede ser automatizar la parte de una tarea de alguien que utilizará el software, para apoyar los procesos del negocio de la organización que ha comisionado el software, a corregir los defectos del software existente, al control de dispositivos, y muchos más. El funcionamiento de los usuarios, los procesos del negocio, y los dispositivos es típicamente complejo. Por extensión, por lo tanto, los requisitos de software son típicamente una combinación compleja de requisitos de diversa gente en diversos niveles de una organización y del ambiente en el cual el software funcionará.

**Se clasifican en 2 requisitos**

**Los requisitos funcionales**

describen las funciones que el software va a ejecutar; por ejemplo, ajustarse a un formato de texto o modular una señal. Se conocen también como capacidades.

**requisitos no funcionales**

son los que actúan para obligar la solución. Los requisitos no funcionales se conocen a veces como apremios o requisitos de calidad

**Proceso de los requisitos**

Esta sección introduce el proceso de los requisitos del software, orientando las cinco subzonas restantes y demostrando cómo el proceso de los requisitos encaja con el proceso de ingeniería del software.

**Captura de los requisites**

Los requisitos tienen muchas fuentes en software típico, y es esencial que todas las fuentes potenciales estén identificadas y evaluadas para su impacto en él. Este asunto se diseña para promover el conocimiento de las varias fuentes de los requisitos del software y de los armazones para manejarlos.

**Análisis de requisitos**

Este asunto se refiere al proceso de analizar requisitos para:

- Detectar y resolver los conflictos entre los requisitos

- Descubrir los límites del software y cómo debe obrar recíprocamente con su ambiente

- Elaborar los requisitos del sistema para derivar requisitos software

La vista tradicional del análisis de requisitos ha sido que esté reducida a modelado conceptual utilizando uno de varios métodos de análisis tales como Análisis Estructurados y Técnicas de Diseño (SADT). Mientras que el modelado conceptual es importante, nosotros incluimos la clasificación de requisitos para ayudar a informar a compensaciones entre los requisitos (clasificación de los requisitos) y el proceso de establecer estas compensaciones (negociación de los requisitos)

**Especificación de requisitos**

Para la mayoría de las profesiones de la ingeniería, el término especificación se refiere a la asignación de valores o límites numéricos para metas del diseño del producto. Los sistemas físicos típicos tienen un número relativamente pequeño de tales valores. Típicamente el software tiene una gran cantidad de requisitos, y el énfasis se comparte entre la ejecución de la cuantificación numérica y el manejo de la complejidad de la interacción entre el gran número de requisitos. Así pues, en software el término, especificación de requisitos del software se refiere típicamente a la producción de un documento, o a su equivalente electrónico, que puede estar sistemáticamente repasado, evaluado, y aprobado. Para los sistemas complejos, particularmente ésos que implican componentes no software, se elaboran tres tipos de documentos: definición de sistema, sistema requisitos, y requisitos del software. Para sistemas simples, solamente el tercero de éstos es requerido. Los tres documentos se describen aquí, entendiendo que combinados pueden ser apropiados.

**Validación de los requisitos**

Los documentos de los requisitos pueden estar conformes a la validación y procedimientos de verificación. Los requisitos pueden ser validados para asegurarse de que el ingeniero del software entiende los requisitos, y es también importante para verificar que un documento de requisitos se conforma con la compañía de los estándares, y éste es comprensible, constante, y finito. Las notaciones formales ofrecen la ventaja importante de permitir que las dos características pasadas sean probadas (en un sentido estricto, por lo menos). Diversos stakeholders, incluyendo los representantes del cliente y del revelador, deben también repasar los documentos. Los documentos de los requisitos son conformes a las mismas prácticas de gerencia de la configuración del software como los otros puntos relevantes de los procesos del ciclo de vida del software.