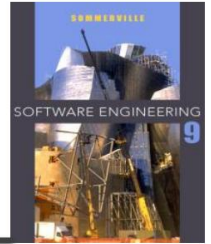


Capítulo 1 - Introducción

Lección 1



Temas tratados

Desarrollo de software profesional

- ¿Qué se entiende por ingeniería de software?

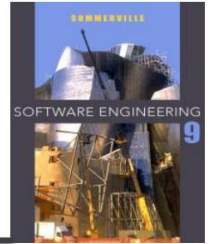
Ética de la ingeniería de software

- Una breve introducción a los problemas éticos que afectan a la ingeniería de software.

Estudios de caso

- Una introducción a tres ejemplos que se utilizan en capítulos posteriores en el libro.

Ingeniería de software



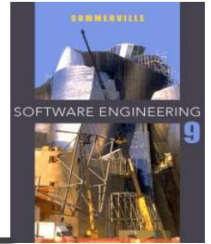
Las economías de TODAS las naciones desarrolladas son depende del software.

Cada vez más sistemas están controlados por software.

La ingeniería de software se ocupa de las teorías, métodos y herramientas para el desarrollo de software profesional.

El gasto en software representa una fracción significativa del PNB en todos los países desarrollados.

Costos del software

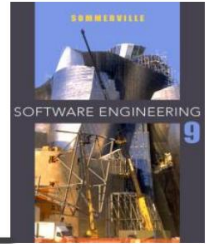


Los costos del software a menudo dominan los costos del sistema informático.
Los costos del software en una PC suelen ser mayores que el costo del hardware.

El mantenimiento del software cuesta más que su desarrollo.

Para sistemas con una vida útil prolongada, los costos de mantenimiento pueden ser varias veces mayores que los costos de desarrollo.

La ingeniería de software se ocupa del desarrollo de software rentable.



Productos de software

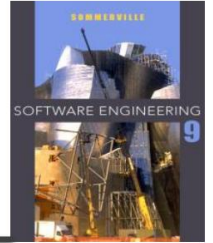
Productos genéricos

- Sistemas autónomos que se comercializan y venden a cualquier cliente que desee comprarlos.
- Ejemplos: software de PC, como programas de gráficos, proyectos herramientas de gestión; software CAD; software para mercados específicos como sistemas de citas para dentistas.

Productos personalizados

- Software encargado por un cliente específico para cumplir sus propias necesidades.
- Ejemplos: sistemas de control integrados, software de control de tráfico aéreo, sistemas de monitoreo de tráfico.

Especificaciones del producto



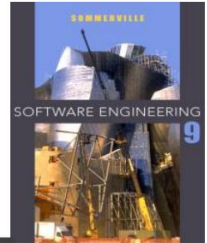
Productos genéricos

- La especificación de lo que debe hacer el software es propiedad del desarrollador del software y las decisiones sobre cambios en el software las toma el desarrollador.

Productos personalizados

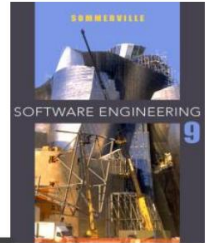
- La especificación de lo que debe hacer el software es propiedad del cliente y es él quien toma las decisiones sobre los cambios que se requieren.

Preguntas frecuentes sobre ingeniería de software

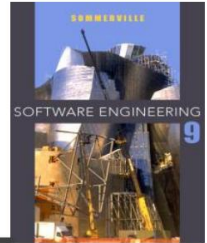


Pregunta	Respuesta
¿Qué es el software?	Programas de ordenador y documentación asociada. Los productos de software pueden desarrollarse para un cliente específico o para un mercado general.
¿Cuáles son los atributos de un buen software?	Un buen software debe ofrecer la funcionalidad y el rendimiento requeridos por el usuario y debe ser mantenible, confiable y utilizable.
¿Qué es la ingeniería de software?	La ingeniería de software es una disciplina de ingeniería que se ocupa de todos los aspectos de la producción de software.
¿Cuáles son las actividades fundamentales de la ingeniería de software?	Especificación de software, desarrollo de software, validación de software y evolución de software.
¿Cuál es la diferencia entre ingeniería de software y ciencia informática?	La informática se centra en la teoría y los fundamentos; la ingeniería de software se ocupa de los aspectos prácticos del desarrollo y la entrega de software útil.
¿Cuál es la diferencia entre ingeniería de software e ingeniería de sistemas?	La ingeniería de sistemas se ocupa de todos los aspectos del desarrollo de sistemas informáticos, incluidos la ingeniería de hardware, software y procesos. La ingeniería de software forma parte de este proceso más general.

Preguntas frecuentes sobre ingeniería de software

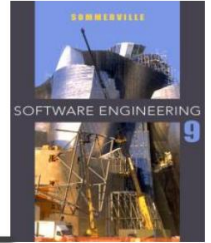


Pregunta	Respuesta
¿Cuáles son los desafíos clave que enfrenta la ingeniería de software?	Hacer frente a la creciente diversidad, a las demandas de plazos de entrega reducidos y al desarrollo de software confiable.
¿Cuáles son los costos de la ingeniería de software?	Aproximadamente el 60% de los costos de software son costos de desarrollo y el 40% son costos de prueba. En el caso del software personalizado, los costos de evolución a menudo superan los costos de desarrollo.
¿Cuáles son las mejores técnicas y métodos de ingeniería de software?	Si bien todos los proyectos de software deben ser gestionados y desarrollados por profesionales, existen distintas técnicas adecuadas para distintos tipos de sistemas. Por ejemplo, los juegos siempre deben desarrollarse utilizando una serie de prototipos, mientras que los sistemas de control críticos para la seguridad requieren el desarrollo de una especificación completa y analizable. Por tanto, no se puede decir que un método sea mejor que otro.
¿Qué diferencias ha introducido la web en la ingeniería de software?	La Web ha permitido disponer de servicios de software y la posibilidad de desarrollar sistemas basados en servicios altamente distribuidos. El desarrollo de sistemas basados en la Web ha propiciado avances importantes en los lenguajes de programación y la reutilización de software.



Atributos esenciales de un buen software

Característica del producto Descripción	
Mantenibilidad	El software debe estar escrito de tal manera que pueda evolucionar. Satisfacer las necesidades cambiantes de los clientes. Este es un atributo fundamental. Porque el cambio de software es un requisito inevitable de un entorno empresarial cambiante.
Confiabilidad y seguridad	La confiabilidad del software incluye una variedad de características Incluyendo confiabilidad, seguridad y protección. Software confiable no debe causar daños físicos o económicos en caso de Fallo del sistema. Los usuarios malintencionados no deberían poder acceder o dañar el sistema.
Eficiencia	El software no debe hacer un uso desperdiciado de los recursos del sistema, como como ciclos de memoria y procesador. Por lo tanto, la eficiencia incluye capacidad de respuesta, tiempo de procesamiento, utilización de la memoria, etc.
Aceptabilidad	El software debe ser aceptable para el tipo de usuarios a los que va dirigido. diseñado. Esto significa que debe ser comprensible, utilizable y compatible con otros sistemas que utilicen.



Ingeniería de software

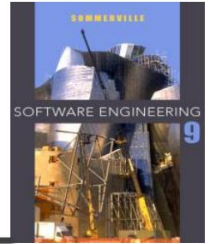
La ingeniería de software es una disciplina de ingeniería que se ocupa de todos los aspectos de la producción de software desde las primeras etapas de la especificación del sistema hasta el mantenimiento del sistema después de su uso.

Disciplina de ingeniería

- Utilizar teorías y métodos apropiados para resolver problemas teniendo en cuenta las limitaciones organizacionales y financieras.

Todos los aspectos de la producción de software

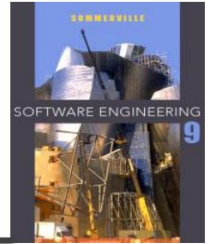
- No sólo el proceso técnico de desarrollo. También el proyecto. gestión y desarrollo de herramientas, métodos, etc. para apoyar la producción de software.



Importancia de la ingeniería de software

Cada vez más, las personas y la sociedad dependen de sistemas de software avanzados. Necesitamos poder producir sistemas fiables y confiables de manera económica y rápida.

A largo plazo, suele ser más económico utilizar métodos y técnicas de ingeniería de software para los sistemas de software que simplemente escribir los programas como si se tratara de un proyecto de programación personal. En la mayoría de los tipos de sistemas, la mayor parte de los costos son los costos de cambiar el software una vez que se ha puesto en uso.



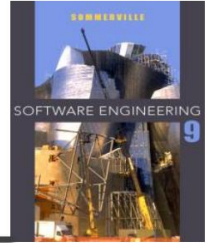
Actividades del proceso de software

Especificación de software, donde los clientes e ingenieros definen el software que se va a producir y las restricciones sobre su funcionamiento.

Desarrollo de software, donde se diseña y programa el software .

Validación de software, donde se verifica que el software es lo que requiere el cliente.

Evolución del software, donde el software se modifica para reflejar los requisitos cambiantes de los clientes y del mercado.



Problemas generales que afectan a la mayoría del software

Heterogeneidad ▪

Cada vez más, se requiere que los sistemas funcionen de manera distribuida. sistemas a través de redes que incluyen diferentes tipos de computadoras y dispositivos móviles.

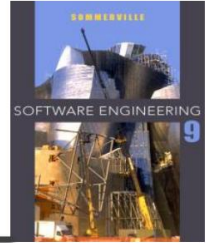
Cambio social y empresarial

- Los negocios y la sociedad están cambiando increíblemente rápido a medida que Las economías emergentes se desarrollan y aparecen nuevas tecnologías. Necesitan poder cambiar su software existente y desarrollar rápidamente software nuevo.

Seguridad y confianza

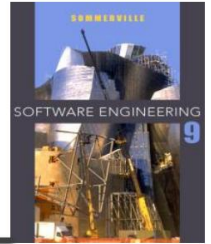
- Como el software está entrelazado con todos los aspectos de nuestras vidas, es Es esencial que podamos confiar en ese software.

Diversidad en la ingeniería de software



Hay muchos tipos diferentes de sistemas de software y no existe un conjunto universal de técnicas de software que sea aplicable a todos ellos.

Los métodos y herramientas de ingeniería de software utilizados Depende del tipo de aplicación que se esté desarrollando, los requisitos del cliente y los antecedentes del equipo de desarrollo.



Tipos de aplicaciones

Aplicaciones independientes

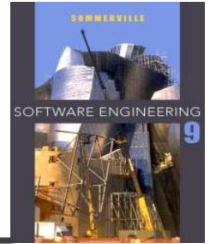
- Son sistemas de aplicación que se ejecutan en una computadora local, como un PC. Incluyen todas las funciones necesarias y no necesitan estar conectados a una red.

Aplicaciones interactivas basadas en transacciones

- Aplicaciones que se ejecutan en una computadora remota y son a los que los usuarios acceden desde sus propios ordenadores o terminales. Entre ellos se encuentran las aplicaciones web, como las de comercio electrónico.

Sistemas de control embebidos ▪ Son

sistemas de control de software que controlan y gestionan dispositivos de hardware. En términos numéricos, probablemente haya más sistemas embebidos que cualquier otro tipo de sistema.



Tipos de aplicaciones

Sistemas de procesamiento por lotes

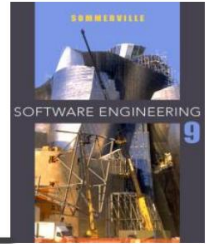
- Son sistemas empresariales diseñados para procesar datos en grandes cantidades. Procesan grandes cantidades de entradas individuales para crear las salidas correspondientes.

Sistemas de entretenimiento

- Se trata de sistemas que son principalmente para uso personal y que Tienen como finalidad entretener al usuario.

Sistemas de modelado y simulación

- Son sistemas que son desarrollados por científicos y ingenieros para modelar procesos o situaciones físicas, que incluyen muchos objetos separados e interactuantes.



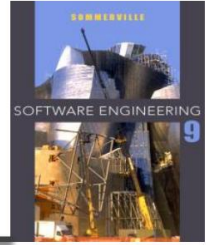
Tipos de aplicaciones

Sistemas de recolección de datos

- Son sistemas que recogen datos de su entorno mediante un conjunto de sensores y envían esos datos a otros sistemas para su

procesamiento. Sistemas de sistemas

- Son sistemas que están compuestos por varios otros sistemas de software.

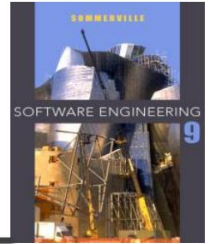


Fundamentos de ingeniería de software

Algunos principios fundamentales se aplican a todos los tipos de sistemas de software, independientemente de las técnicas de desarrollo utilizadas:

- Los sistemas deben desarrollarse mediante un proceso de desarrollo controlado y comprendido. Por supuesto, se utilizan distintos procesos para distintos tipos de software.
- La confiabilidad y el rendimiento son importantes para todo tipo de sistema.
- Es importante comprender y gestionar las especificaciones y los requisitos del software (lo que debe hacer el software). ▪

Cuando sea apropiado, debe reutilizar el software que ya se ha desarrollado en lugar de escribir software nuevo.



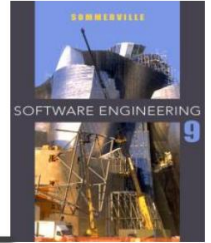
Ingeniería de software y la web

La Web es ahora una plataforma para ejecutar aplicaciones y las organizaciones están desarrollando cada vez más sistemas basados en la Web en lugar de sistemas locales.

Los servicios web (discutidos en el Capítulo 19) permiten
Funcionalidad de la aplicación a la que se puede acceder a través de la web.

La computación en la nube es un enfoque para la provisión de servicios informáticos donde las aplicaciones se ejecutan de forma remota en la "nube".

- Los usuarios no compran software sino que pagan según el uso.



Ingeniería de software web

La reutilización de software es el enfoque dominante para construir sistemas basados en la web.

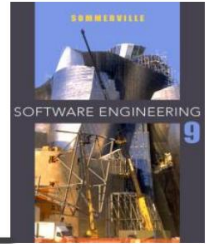
- Al construir estos sistemas, piensa en cómo puedes ensamblarlos a partir de componentes y sistemas de software preexistentes.

Los sistemas basados en la web deben desarrollarse y entregarse de forma incremental.

- En la actualidad se reconoce generalmente que no es práctico especificar todos los requisitos para dichos sistemas de antemano.

Las interfaces de usuario están limitadas por las capacidades de la web. navegadores.

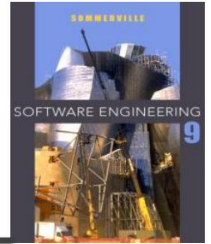
- Las tecnologías como AJAX permiten crear interfaces complejas dentro de un navegador web, pero aún así son difíciles de usar. Los formularios web con secuencias de comandos locales son los más utilizados.



Ingeniería de software basada en la web

Los sistemas basados en la web son sistemas distribuidos complejos, pero los principios fundamentales de la ingeniería de software analizados anteriormente son tan aplicables a ellos como a cualquier otro tipo de sistema.

Las ideas fundamentales de la ingeniería de software, Los conceptos discutidos en la sección anterior se aplican al software basado en web de la misma manera que se aplican a otros tipos de sistemas de software.



Puntos clave

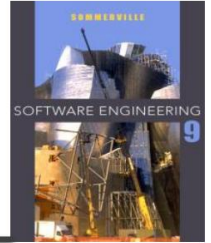
La ingeniería de software es una disciplina de ingeniería que se ocupa de todos los aspectos de la producción de software.

Los atributos esenciales del producto de software son mantenibilidad, confiabilidad, seguridad, eficiencia y aceptabilidad.

Las actividades de alto nivel de especificación, desarrollo, validación y evolución son parte de todos los procesos de software.

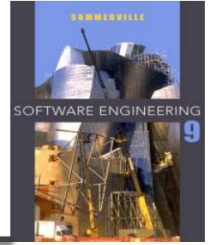
Las nociones fundamentales de la ingeniería de software son universalmente aplicables a todo tipo de desarrollo de sistemas.

Puntos clave



Existen muchos tipos diferentes de sistemas y cada uno requiere herramientas y técnicas de ingeniería de software adecuadas para su desarrollo.

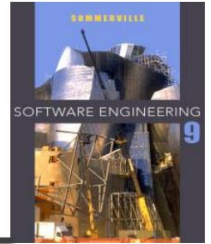
Las ideas fundamentales de la ingeniería de software son aplicables a todos los tipos de sistemas de software.



Capítulo 1 - Introducción

Lección 2

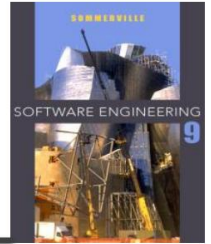
Ética de la ingeniería de software



La ingeniería de software implica responsabilidades más amplias que simplemente la aplicación de habilidades técnicas.

Los ingenieros de software deben comportarse de manera honesta y de manera éticamente responsable si se quiere que sean respetados como profesionales.

El comportamiento ético es más que simplemente respetar la ley; implica seguir un conjunto de principios que son moralmente correctos.



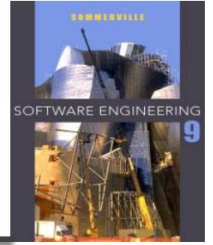
Cuestiones de responsabilidad profesional

Confidencialidad ▪

Los ingenieros normalmente deben respetar la confidencialidad de sus empleadores o clientes, independientemente de si se ha firmado o no un acuerdo de confidencialidad formal.

Competencia ▪

Los ingenieros no deben tergiversar su nivel de competencia.
No deben aceptar a sabiendas un trabajo que esté fuera de su competencia.



Cuestiones de responsabilidad profesional

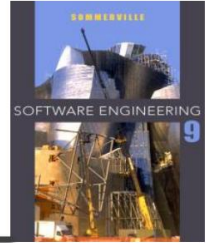
Derechos de propiedad intelectual

- Los ingenieros deben conocer las leyes locales que rigen el uso de propiedad intelectual como patentes, derechos de autor, etc. Deben tener cuidado de garantizar que la propiedad intelectual de los empleadores y clientes esté protegida.

Mal uso de la computadora

- Los ingenieros de software no deben utilizar sus habilidades técnicas para Hacer un uso indebido de los ordenadores de otras personas. El uso indebido de los ordenadores varía desde relativamente trivial (por ejemplo, jugar en el ordenador de un empleador) hasta extremadamente grave (diseminación de virus).

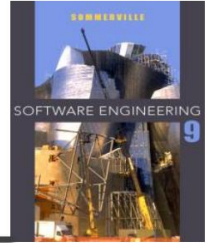
Código de ética de ACM/IEEE



Las sociedades profesionales de los EE. UU. han cooperado para producir un código de práctica ética.

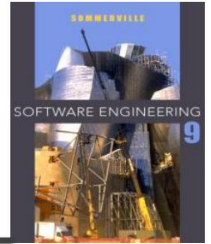
Los miembros de estas organizaciones firman el código de prácticas cuando se unen.

El Código contiene ocho Principios relacionados con el comportamiento y las decisiones que toman los ingenieros de software profesionales, incluidos los profesionales, educadores, gerentes, supervisores y formuladores de políticas, así como los aprendices y estudiantes de la profesión.



Fundamento del código ético

- Las computadoras tienen un papel central y creciente en el comercio, industria, gobierno, medicina, educación, entretenimiento y la sociedad en general. Los ingenieros de software son aquellos que contribuyen mediante participación directa o mediante la enseñanza, al análisis, especificación, diseño, desarrollo, certificación, mantenimiento y prueba de sistemas de software.
- Debido a sus roles en el desarrollo de sistemas de software, los ingenieros de software tienen oportunidades significativas de hacer el bien o causar daño, de permitir que otros hagan el bien o causen daño, o de influenciar a otros para que hagan el bien o causen daño. Para asegurar, tanto como sea posible, que sus esfuerzos se utilizarán para el bien, los ingenieros de software deben comprometerse a hacer de la ingeniería de software una profesión beneficiosa y respetada.



Código de ética de ACM/IEEE

Código de ética y práctica profesional de la ingeniería de software

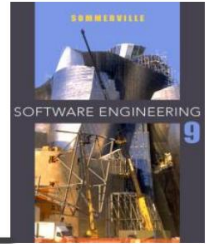
Grupo de trabajo conjunto ACM/IEEE-CS sobre ética y prácticas profesionales en ingeniería de software

PREÁMBULO

La versión corta del código resume las aspiraciones a un alto nivel de abstracción; las cláusulas que se incluyen en la versión completa brindan ejemplos y detalles de cómo estas aspiraciones cambian la forma en que actuamos como profesionales de la ingeniería de software. Sin las aspiraciones, los detalles pueden volverse legalistas y tediosos; sin los detalles, las aspiraciones pueden volverse grandilocuentes pero vacías; juntas, las aspiraciones y los detalles forman un código cohesivo.

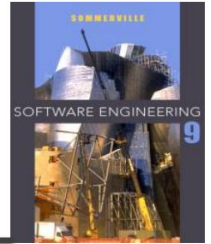
Los ingenieros de software se comprometerán a hacer del análisis, la especificación, el diseño, el desarrollo, la prueba y el mantenimiento de software una profesión beneficiosa y respetada. De conformidad con su compromiso con la salud, la seguridad y el bienestar del público, los ingenieros de software se adherirán a los siguientes ocho principios:

Principios éticos



1. PÚBLICO - Los ingenieros de software deberán actuar en consonancia con el interés público.
2. CLIENTE Y EMPLEADOR - Los ingenieros de software deberán actuar de una manera que sea en el mejor interés de su cliente y empleador, en consonancia con el interés público.
3. PRODUCTO - Los ingenieros de software deberán garantizar que sus productos y las modificaciones relacionadas cumplan con los estándares profesionales más altos posibles.
4. JUICIO - Los ingenieros de software deberán mantener integridad e independencia en su juicio profesional.
5. GESTIÓN - Los gerentes y líderes de ingeniería de software deberán suscribir y promover un enfoque ético para la gestión del desarrollo y mantenimiento de software.
6. PROFESIÓN - Los ingenieros de software deberán promover la integridad y la reputación de la profesión en consonancia con el interés público.
7. COLEGAS - Los ingenieros de software deben ser justos y apoyar a sus colegas.
8. AUTO - Los ingenieros de software participarán en el aprendizaje permanente con respecto al ejercicio de su profesión y promoverán un enfoque ético en el ejercicio de la profesión.

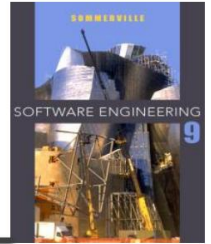
Dilemas éticos



Desacuerdo de principio con las políticas de la alta dirección.

Su empleador actúa de manera poco ética y lanza un sistema crítico para la seguridad sin terminar las pruebas del sistema.

Participación en el desarrollo de armas militares sistemas o sistemas nucleares.



Estudios de caso

Una bomba de insulina personal

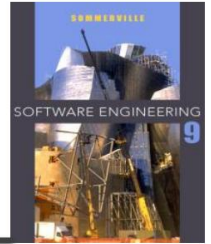
- Un sistema integrado en una bomba de insulina utilizada por diabéticos para mantener el control de la glucosa en sangre.

Un sistema de gestión de casos de salud mental para pacientes

- Un sistema utilizado para mantener registros de las personas que reciben atención
Problemas de salud mental.

Una estación meteorológica en plena naturaleza

- Un sistema de recopilación de datos que recoge datos sobre las condiciones climáticas en áreas remotas.



Sistema de control de bomba de insulina

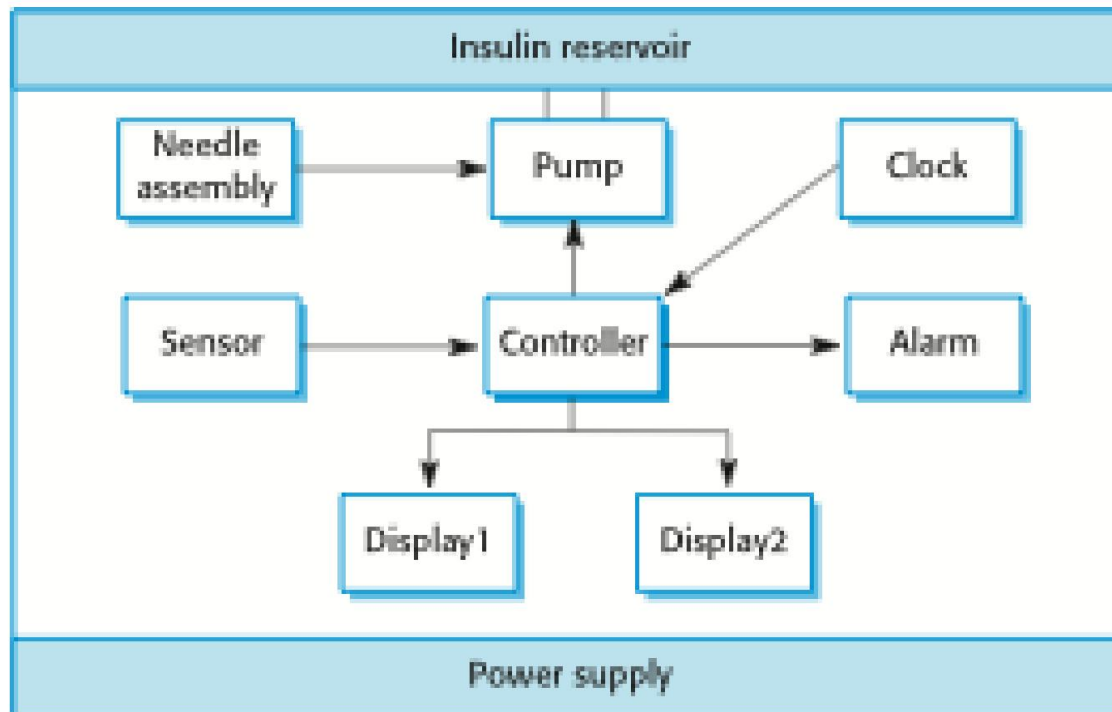
Recopila datos de un sensor de azúcar en sangre y calcula la cantidad de insulina que debe inyectarse.

Cálculo basado en la tasa de cambio del azúcar en sangre. niveles.

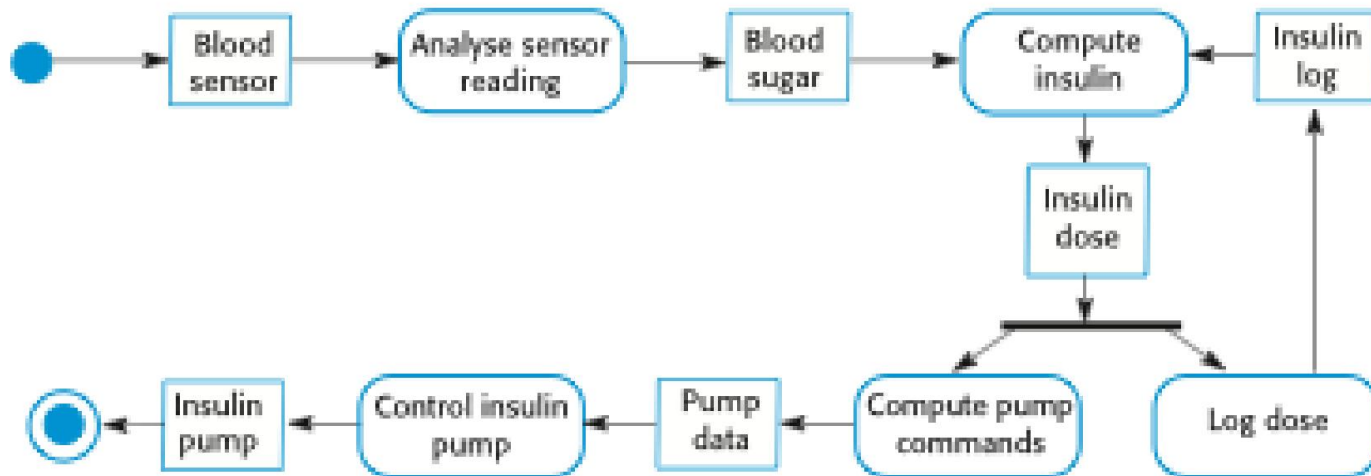
Envía señales a una microbomba para administrar la dosis correcta de insulina.

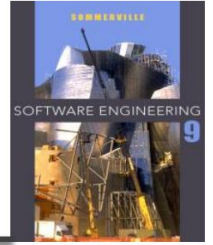
Sistema crítico para la seguridad, ya que los niveles bajos de azúcar en sangre pueden provocar mal funcionamiento del cerebro, coma y muerte; los niveles altos de azúcar en sangre tienen consecuencias a largo plazo, como daños en los ojos y los riñones.

Arquitectura del hardware de la bomba de insulina



Modelo de actividad de la bomba de insulina





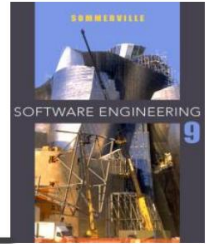
Requisitos esenciales de alto nivel

El sistema deberá estar disponible para administrar insulina cuando requerido.

El sistema deberá funcionar de manera confiable y suministrar la cantidad correcta de insulina para contrarrestar el nivel actual de insulina en la sangre. azúcar.

Por lo tanto, el sistema debe diseñarse y implementado para garantizar que el sistema siempre cumpla con estos requisitos.

Un sistema de información para pacientes de salud mental cuidado

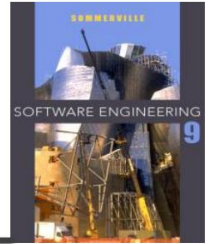


Un sistema de información de pacientes para apoyar la atención de salud mental es un sistema de información médica que mantiene información sobre los pacientes que sufren problemas de salud mental y los tratamientos que han recibido.

La mayoría de los pacientes con problemas de salud mental no requieren un tratamiento hospitalario específico, sino que necesitan asistir periódicamente a clínicas especializadas donde pueden consultar a un médico que tiene un conocimiento detallado de sus problemas.

Para facilitar la asistencia de los pacientes, estas clínicas no sólo se llevan a cabo en hospitales, sino que también pueden realizarse en consultorios médicos locales o centros comunitarios.

MHC-PMS



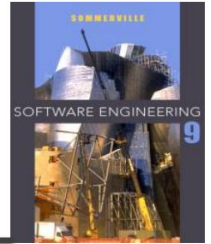
El MHC-PMS (Atención de Salud Mental-Paciente)

Sistema de Gestión de Riesgos (SGRR) es un sistema de información que está destinado a ser utilizado en clínicas.

Hace uso de una base de datos centralizada de pacientes información, pero también ha sido diseñado para ejecutarse en una PC, de modo que se pueda acceder a ella y utilizarla desde sitios que no tienen conectividad de red segura.

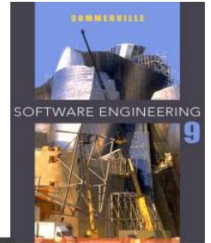
Cuando los sistemas locales tienen acceso seguro a la red, Utilizan información del paciente en la base de datos, pero pueden descargar y utilizar copias locales de los registros del paciente cuando están desconectados.

Objetivos del MHC-PMS

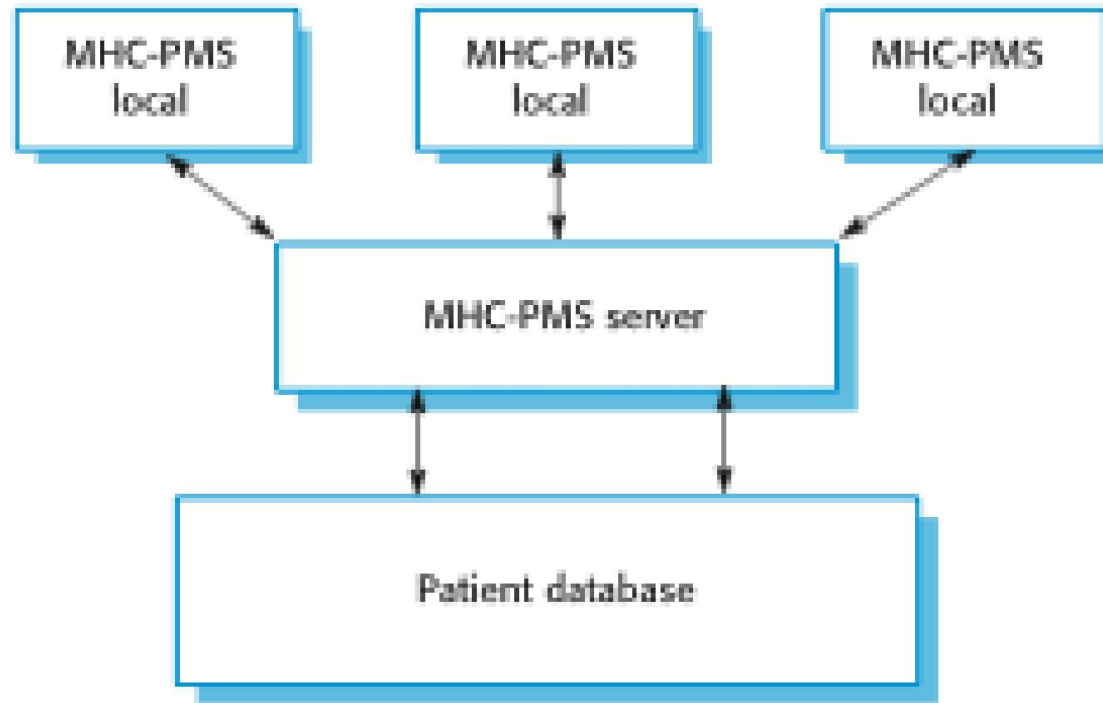


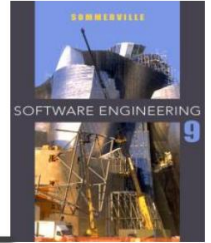
Generar información de gestión que permita a los gerentes de los servicios de salud evaluar su desempeño frente a los objetivos locales y gubernamentales.

Proporcionar al personal médico información oportuna para apoyar el tratamiento de los pacientes.



La organización del MHC-PMS





Características principales del MHC-PMS

Gestión de atención individual ▪ Los

médicos pueden crear registros para pacientes, editar la información en el sistema, ver el historial del paciente, etc. El sistema admite resúmenes de datos para que los médicos puedan conocer rápidamente los problemas y tratamientos clave que se han prescrito.

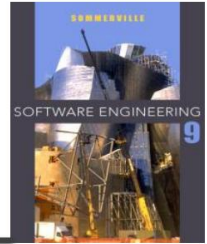
Monitoreo de pacientes

- El sistema monitorea los registros de los pacientes que están involucrados en tratamiento y emite advertencias si se detectan posibles problemas.

Informes administrativos

- El sistema genera informes mensuales de gestión que muestran el número de pacientes atendidos en cada clínica, el número de pacientes que han entrado y salido del sistema de atención, el número de pacientes internados, los medicamentos prescritos y sus costos, etc.

Preocupaciones sobre MHC-PMS

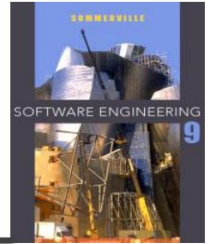


Privacidad

- Es esencial que la información del paciente sea confidencial y nunca se divulgue a nadie excepto al personal médico autorizado y al propio paciente.

Seguridad

- Algunas enfermedades mentales hacen que los pacientes tengan tendencias suicidas o se conviertan en un peligro para otras personas. Siempre que sea posible, el sistema debe advertir al personal médico sobre pacientes potencialmente suicidas o peligrosos.
- El sistema debe estar disponible cuando sea necesario, de lo contrario, la seguridad puede verse comprometida y puede ser imposible prescribir la medicación correcta a los pacientes.

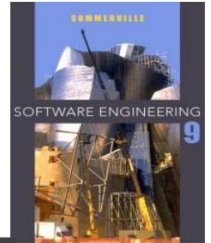


Estación meteorológica en la naturaleza

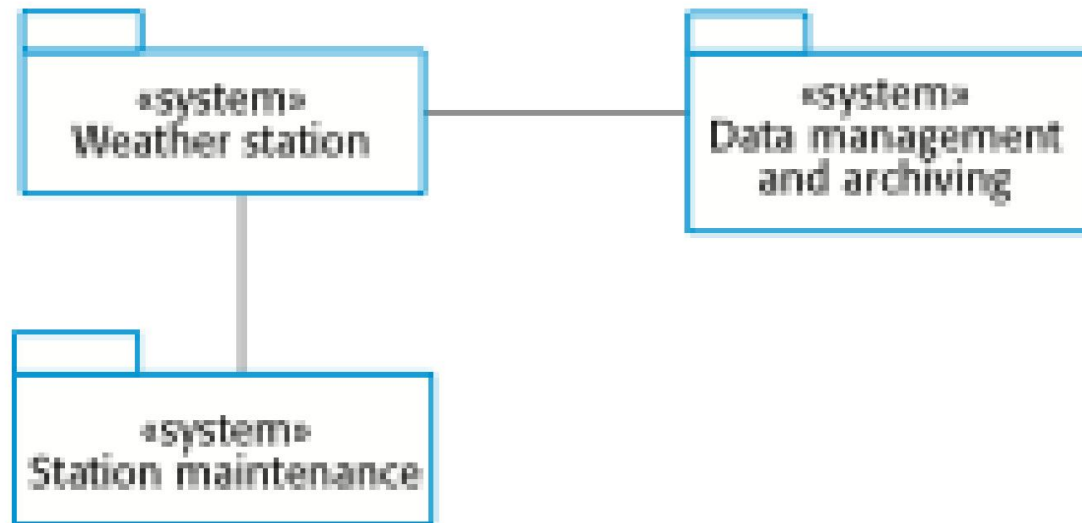
El gobierno de un país con grandes áreas de
La región desértica decide desplegar varios cientos de estaciones
meteorológicas en zonas remotas.

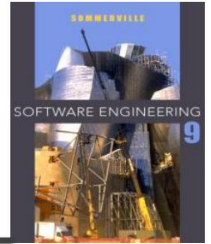
Las estaciones meteorológicas recogen datos de un conjunto de instrumentos
que miden la temperatura y la presión, la luz solar, las
precipitaciones y la velocidad y dirección del viento.

- La estación meteorológica incluye una serie de instrumentos que miden
parámetros meteorológicos como la velocidad y dirección del viento, la
temperatura del suelo y del aire, la presión barométrica y la pluviosidad
durante un período de 24 horas. Cada uno de estos instrumentos está
controlado por un sistema de software que toma lecturas periódicas de
los parámetros y gestiona los datos recopilados por los instrumentos.



El entorno de la estación meteorológica





Sistema de información meteorológica

El sistema de estación meteorológica

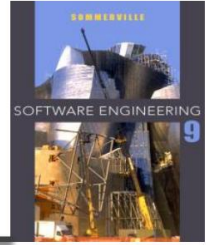
- Es responsable de recopilar datos meteorológicos, realizar un procesamiento inicial de datos y transmitirlos al sistema de gestión de datos.

El sistema de gestión y archivo de datos

- Este sistema recopila los datos de todas las condiciones meteorológicas de la naturaleza. estaciones, realiza el procesamiento y análisis de datos y archiva los datos.

El sistema de mantenimiento de la estación

- Este sistema puede comunicarse vía satélite con todas las áreas silvestres. estaciones meteorológicas para monitorear la salud de estos sistemas y proporcionar informes de problemas.

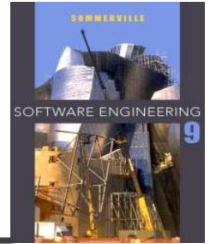


Funcionalidad de software adicional

Supervisar los instrumentos, la energía y la comunicación. hardware y reportar fallas al sistema de gestión.

Gestionar la energía del sistema, garantizando que las baterías se carguen siempre que las condiciones ambientales lo permitan, pero también que los generadores se apaguen en condiciones climáticas potencialmente dañinas, como vientos fuertes.

Apoyar la reconfiguración dinámica donde partes del software se reemplazan con nuevas versiones y donde los instrumentos de respaldo se conectan al sistema en caso de falla del sistema.



Puntos clave

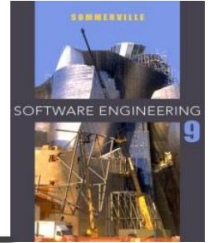
Los ingenieros de software tienen responsabilidades con el
La profesión de ingeniería y la sociedad no deberían preocuparse sólo por cuestiones técnicas.

Las sociedades profesionales publican códigos de conducta que establecen los estándares de comportamiento que se esperan de sus miembros.

En el libro se utilizan tres estudios de caso:

- Un sistema de control de bomba de insulina integrado
- Un sistema para la gestión de pacientes de atención de salud mental
- Una estación meteorológica en plena naturaleza

Estructura y organización del curso



Agregue aquí su propio material sobre cómo llevará a cabo el curso.