

EXAMEN CORTO PRACTICO # 1 DISEÑO DE SOFTWARE

Nelson Rodríguez Zúñiga
e-mail: nhrodriguez@gmail.com
II Parte Preguntas

1. ¿Qué es un sistema de información?

Es un conjunto de elementos orientados al tratamiento y la administración de datos e información, generados para cubrir una necesidad u objetivo, estos elementos formaran parte de las siguientes categorías:

Personas.

Actividades o técnicas de trabajo.

Recursos materiales en general (recursos informáticos y de comunicación)

Los sistemas de información es un concepto muy genérico, dependiendo de para qué se vaya a usar así varia de su función.

Por ejemplo, en geografía y cartografía un sistema de información se utiliza para almacenar, editar, analizar, compartir y desplegar información georreferenciada.

En informática, es cualquier sistema de información que se utilice para obtener, almacenar, administrar, controlar, procesar, transmitir y recibir datos.

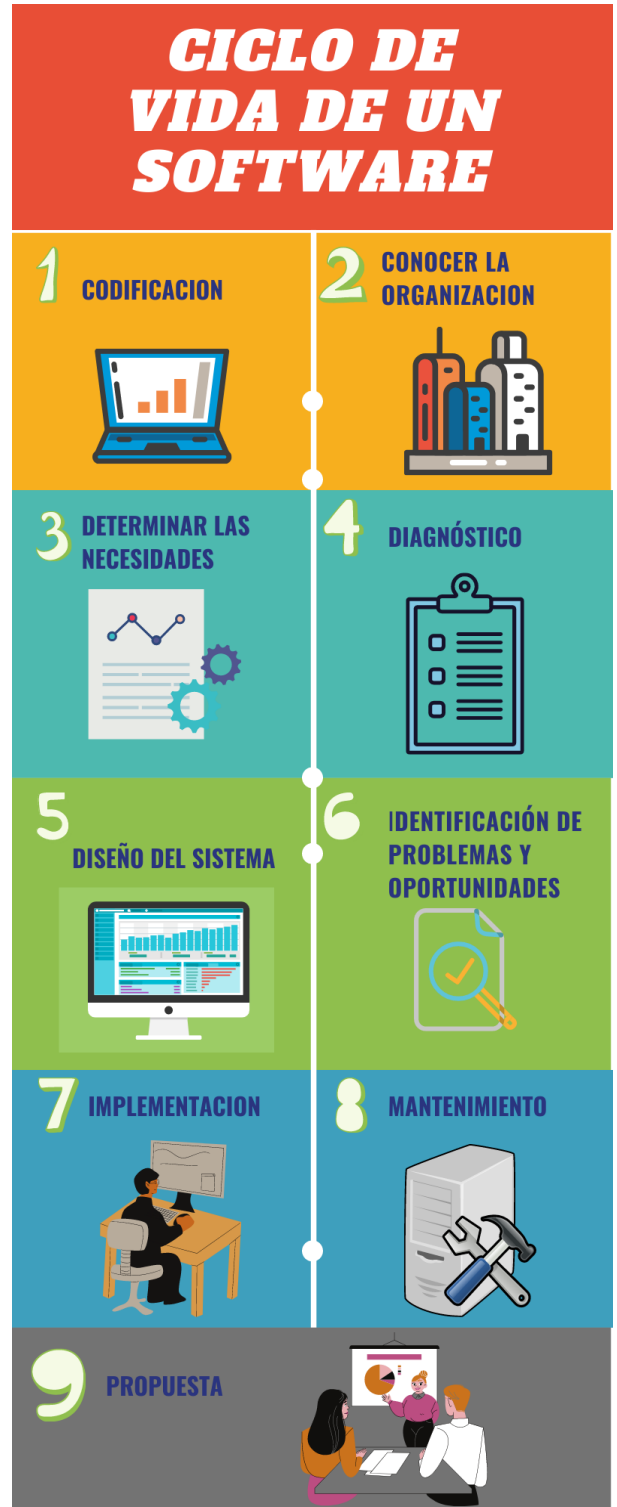
2. ¿Cuáles son las 4 actividades que producen información para el análisis y toma de decisiones?

- I. Recopilación de Datos: captura y recolecta datos en bruto tanto del interior como del exterior de la organización.
- II. Almacenamiento: Guardar de forma estructurada la información recopilada.
- III. Procesamiento: Convierte esa entrada de datos en una forma más significativa.
- IV. Distribución: transfiere la información procesada a las personas o roles que la usarán.

3. Creé un infográfico del Ciclo de vida del Software. (Para ello puede usar el siguiente enlace https://www.canva.com/es_es/)

A continuación, se presenta el siguiente Infográfico con el ciclo de vida de un software.

Donde conoceremos las partes por el cual debe pasar todo el estudio, diagnostico y procesos antes de llegar a una etapa de implementación y puesta en marcha. Este proceso es importante verlo y esta resumido en la figura que a continuación se presenta.



4. ¿Qué es ingeniería del software?

La ingeniería de software es una especialidad que consiste en sistemas, instrumentos y técnicas que se emplean en el desarrollo de los programas informáticos. La ingeniería de software, también, incorpora el análisis precedente de la situación, el bosquejo del proyecto, el desarrollo del software, el ensayo necesario para comprobar su funcionamiento correcto y poner en funcionamiento el sistema.

En la actualidad el software tiene un doble papel. Es el producto, pero al mismo tiempo, actúa como el conductor que entrega el producto. Como conductor utilizado para entregar el producto, actúa como base de control, por ejemplo, un sistema operativo, o un sistema gestor de redes.

El software actúa como distribuidor y hace llegar a los usuarios, el producto más importante del este siglo: la información.

5. ¿Cuál es la importancia de un correcto levantado de requisitos?

Los requerimientos son importantes debido a que son el hilo conductor de todo desarrollo de software. Obtener requerimientos de calidad demuestra que el trabajo realizado culminará con éxito, esto se debe a dos factores:

1. La utilización adecuada de las técnicas de captura de requerimientos con los clientes.

2. Las experiencias de los analistas del proyecto.

Esto sucede porque la experiencia de trabajo en el rol le permite al equipo de Analistas del Proyecto establecer que técnicas van a utilizar a la hora de la entrevista con el cliente debido a que los clientes no entienden el lenguaje informático, es por eso que se debe tener en cuenta el lenguaje el cual se va a aplicar a la hora de la entrevista con el cliente.

6. ¿Cuáles son los tipos de sistemas de información?

1. **SISTEMA DE PROCESAMIENTO DE TRANSACCIONES:** Son aquellos sistemas que recopilan, almacenan y alteran la información que ha sido recopilada a través de transacciones que se llevaron a cabo dentro de una empresa. Estos sistemas existen con la finalidad de procesar transacciones diarias, acumulando toda la información recibida en una base de datos para su posterior consulta. Es un sistema que soporta las operaciones diarias de un negocio.
2. **SISTEMA DE INFORMACIÓN GERENCIAL:** Este tipo de sistema de información tiene como objetivo otorgar información a la administración de la empresa, para que pueda resolver problemas a través de la interacción entre tecnologías y personas. Los datos aportados por el sistema deben disponer de cuatro cualidades elementales: calidad, oportunidad, cantidad y relevancia. El Sistema de Información Gerencial, es un sistema de apoyo

3. a la gerencia, que ayuda al equipo directivo a tomar decisiones en situaciones en las que existe incertidumbre sobre los posibles resultados o consecuencias.

4. **SISTEMA DE SOPORTE A DECISIONES:** Es una herramienta de Business Intelligence diseñada para analizar los datos de una organización. Generalmente, los informes que entregan estos sistemas, son predefinidos (es decir que pocas veces permiten profundizar en los datos, navegar en ellos, etc., lo que no hace su lectura una tarea sencilla). De igual manera, los resultados arrojados siempre ofrecen la posibilidad (gracias a toda la información que detalla el sistema) de optar por la decisión más acertada. Cabe destacar que el sistema en cuestión, se basa en el estudio y la comparación entre un conjunto de variables con el objeto de contribuir a la toma de decisiones dentro de una empresa. El apoyo dado por el sistema involucra la estimación, valoración y balance entre alternativas.

5. **SISTEMA DE INFORMACIÓN EJECUTIVA:** Este sistema también está dirigido a la gerencia de una empresa; posibilita a los altos mandos acceso a la información interna y externa de la compañía, lo que les facilitará la tarea de conocer el estado de todos los indicadores, incluso aquellos que no cumplan con las expectativas y a partir de esto, tomar las medidas que considere adecuadas. Estos sistemas, generalmente ofrecen la información en gráficos, pero con la posibilidad de acceder a datos básicos más detallados si fuera necesario.

6. **SISTEMAS DE CONTROL DE PROCESOS:** Los sistemas de control de procesos de negocio (BPM por sus siglas en inglés) son aquellos diseñados para monitorizar y controlar los procesos industriales o físicos, como, por ejemplo, sería el caso de la refinación de petróleo, generación de energía o los sistemas de producción de acero en una planta siderúrgica.

7. **SISTEMAS DE COLABORACIÓN EMPRESARIAL:** Es uno de los tipos de sistemas de información más utilizados (ERP por sus siglas en inglés). Su objetivo es ayudar a los directivos de una empresa a controlar el flujo de información en sus organizaciones. Estos sistemas no son específicos de un nivel concreto en la empresa, ya que proporcionan un soporte importante para una amplia gama de usuarios. Estos sistemas de información están diseñados para soportar tareas de oficina como sistemas multimedia, correos electrónicos, videoconferencias y transferencias de archivos.

7. ¿Defina lo siguiente CRM-ERP-SCM?

CRM (Costumer Relationship Management)

Es una estrategia de negocio orientada a la fidelización de clientes.

Permite a todos los empleados de una empresa disponer de información actualizada sobre los mismos, con el objetivo de optimizar la relación entre empresa y cliente. Además, ayuda a conocer todos los diferentes puntos de contacto con los cuales el cliente interactúa en la empresa.

CRM es un modelo de gestión basada en la satisfacción del cliente, el concepto más cercano es el de Marketing relacional.

A nivel de software para la administración de la relación con los clientes.

Sistemas informáticos de apoyo a la gestión de las relaciones con los clientes, a la venta y al marketing. 2 Dicho software puede comprender varias funcionalidades para gestionar las ventas y los clientes de la empresa: automatización y promoción de ventas, tecnologías data warehouse (almacén de datos) para agregar la información transaccional y proporcionar capa de reporting, dashboards e indicadores claves de negocio, funcionalidades para seguimiento de campañas de marketing y gestión de oportunidades de negocio, capacidades predictivas y de proyección de ventas.

La parte fundamental de toda estrategia CRM es la base de datos, con la cual se puede realizar un seguimiento constante a los clientes. Actualmente la manera innovadora que han encontrado los almacenes de cadena para realizar seguimiento a sus clientes, es por medio de las tarjetas de puntos que otorgan regalos. De esta manera, por medio de la tarjeta, se puede saber en promedio cada cuánto el cliente visita el almacén y que productos suele adquirir. Uno de los mayores problemas para que las empresas exploten un producto CRM es el alto costo de estos productos comerciales, licencias adicionales como un sistema operativo y más aún el alto costo de la su implantación y puesta en marcha, afortunadamente existen también diversos proyectos de sistemas CRM con licencia

ERP (Enterprice Resource Plannig)

Sistema de planificación de recursos, su objetivo es la planificación de los recursos de la organización, es tener claramente identificado como llegar a los productos finales desde la materia prima, o sea, desde el inventario e insumos para poder determinar la cantidad que llegaremos a generar de productos finales, integran la información y los procesos de la organización en un solo sistema.

Son sistemas de gestión automatizan algunas prácticas de negocio asociadas con los aspectos operativos y productivos de la organización.

Objetivos principales de ERP:

Optimización de procesos empresariales.

Acceso a información de forma confiable, verás y a tiempo.

La posibilidad de compartir información con las demás áreas de la organización buscando integridad y buen desarrollo de los procesos.

Eliminación procesos incensarios dentro de la compañía.

SCM (Supply Chain Management)

O administración de cadena de suministros, se define como la unión de proveedores y consumidores por medio de una cadena. Se busca principalmente que los procesos que añaden más valor a la cadena, estén integrados para evitar disconformidades en uno u otro.

El modelo SCM, se toma cada vez con mayor interés por las empresas, debido a su alta efectividad. El objetivo de las empresas se deriva principalmente de dos factores: la globalización de mercados y producción y la evolución tecnológica.

La globalización de los mercados y la producción ha generado una voraz competencia mundial entre empresas, a la competencia se une el avance de muchas industrias por el intercambio tecnológico, factor que hace más dura la competencia. Esto ha generado un ciclo de vida cada vez más corto para los productos.

La innovación tecnológica, de información, comunicación y de transporte ha permitido el desarrollo de métodos y herramientas cada vez eficaces para sincronizar la oferta y la demanda, por medio de la coordinación entre las actividades desarrolladas por proveedores, productores y distribuidores.

8. ¿Qué es calidad de software?

La calidad en ingeniería del software es el cumplimiento de los requerimientos contractuales por parte del producto software desarrollado, así como durante el proceso de desarrollo. La calidad se obtiene mejorando día a día el proceso de producción, mantenimiento y gestión del software

9. Mencione y explique los distintos modelos para el desarrollo de software.

Modelo de Prototipo:

Pertenece a los modelos de desarrollo evolutivo. Se debe de construir en poco tiempo usando programa adecuados y optimizando recursos.

Este diseño se centra en una representación de la parte visual del software que serán vistos por el usuario final. Este será evaluado por el cliente para retroalimentación por este medio se refinan los requisitos del sistema que se desarrollará. La interacción ocurre cuando el prototipo se ajusta para satisfacer las necesidades del cliente. Esto permite que al mismo tiempo el desarrollador entienda mejor lo que se debe hacer y el cliente vea resultados a corto plazo.

Etapas:

Plan rápido.

Modelo, diseño rápido.

Construcción del prototipo.

Desarrollo, entrega y retroalimentación

Comunicación.

Entrega del desarrollo final.

Ventajas:

Este modelo es útil cuando el cliente conoce los objetivos generales para el software, pero no identifica

los requisitos detallados de entrada, procesamiento o salida.

También ofrece un mejor enfoque cuando el responsable del desarrollo del software está inseguro de la eficacia de un algoritmo, de la adaptabilidad de un sistema operativo o de la forma que debería tomar la interacción humano-máquina

Se puede reutilizar el código.

La construcción de prototipos se puede utilizar como un modelo del proceso independiente, se emplea más comúnmente como una técnica susceptible de implementarse dentro del contexto de cualquiera de los modelos del proceso expuestos. Sin importar la forma en que éste se aplique, el paradigma de construcción de prototipos ayuda al desarrollador de software y al cliente a entender de mejor manera cuál será el resultado de la construcción cuando los requisitos estén satisfechos. De esta manera, este ciclo de vida en particular, involucra al cliente más profundamente para adquirir el producto.

Desventajas:

El usuario tiende a crearse unas expectativas cuando ve el prototipo de cara al sistema final. A causa de la intención de crear un prototipo de forma rápida, se suelen desatender aspectos importantes, tales como la calidad y el mantenimiento a largo plazo, lo que obliga en la mayor parte de los casos a reconstruirlo una vez que el prototipo ha cumplido su función. Es frecuente que el usuario se muestre reacio a ello y pida que sobre ese prototipo se construya el sistema final, lo que lo convertiría en un prototipo evolutivo, pero partiendo de un estado poco recomendado.

En aras de desarrollar rápidamente el prototipo, el desarrollador suele tomar algunas decisiones de implementación poco convenientes (por ejemplo, elegir un lenguaje de programación incorrecto porque proporcione un desarrollo más rápido). Con el paso del tiempo, el desarrollador puede olvidarse de la razón que le llevó a tomar tales decisiones, con lo que se corre el riesgo de que dichas elecciones pasen a formar parte del sistema final.

Modelo Espiral:

El desarrollo en espiral es un modelo de ciclo de vida del software definido por primera vez por Barry Boehm en 1986. Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades. Las actividades no están fijadas a ninguna prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior.

Básicamente consiste en una serie de ciclos que se repiten en forma de espiral, comenzando desde el centro. Se suele interpretar como que dentro de cada ciclo de la espiral se sigue un Modelo Cascada, pero no necesariamente debe ser así. El Espiral puede verse como un modelo evolutivo que conjuga la naturaleza iterativa del modelo MCP con los aspectos controlados y sistemáticos del Modelo Cascada, con el agregado de gestión de riesgo.

Ciclos o Iteraciones

En cada vuelta o iteración hay que tener en cuenta:

Los Objetivos: qué necesidad debe cubrir el producto.

Alternativas: las diferentes formas de conseguir los objetivos de forma exitosa, desde diferentes puntos de vista como pueden ser:

Características: experiencia del personal, requisitos a cumplir, etc.

Formas de gestión del sistema.

Riesgo asumido con cada alternativa.

Desarrollar y Verificar: Programar y probar el software.

Si el resultado no es el adecuado o se necesita implementar mejoras o funcionalidades:

Se planificarán los siguientes pasos y se comienza un nuevo ciclo de la espiral. La espiral tiene una forma de caracola y se dice que mantiene dos dimensiones, la radial y la angular:

Angular: Indica el avance del proyecto del software dentro de un ciclo.

Radial: Indica el aumento del coste del proyecto, ya que con cada nueva iteración se pasa más tiempo desarrollando.

Este sistema es muy utilizado en proyectos grandes y complejos como puede ser, por ejemplo, la creación de un Sistema Operativo.

Al ser un modelo de Ciclo de Vida orientado a la gestión de riesgo se dice que uno de los aspectos fundamentales de su éxito radica en que el equipo que lo aplique tenga la necesaria experiencia y habilidad para detectar y catalogar correctamente los riesgos.

Tareas

Para cada ciclo habrá cuatro actividades:

Determinar Objetivos.

Análisis del riesgo.

Desarrollar y probar.

Planificación.



Determinar o fijar objetivos

Fijar también los productos definidos a obtener: requisitos, especificación, manual de usuario.

Fijar las restricciones.

Identificación de riesgos del proyecto y estrategias alternativas para evitarlos.

Hay una cosa que solo se hace una vez: planificación inicial.

Desarrollar, verificar y validar (probar)

Tareas de la actividad propia y de prueba.

Análisis de alternativas e identificación resolución de riesgos.

Dependiendo del resultado de la evaluación de los riesgos, se elige un modelo para el desarrollo, el que

puede ser cualquiera de los otros existentes, como formal, evolutivo, cascada, etc. Así si por ejemplo si los riesgos en la interfaz de usuario son dominantes, un modelo de desarrollo apropiado podría ser la construcción de prototipos evolutivos. Si lo riesgos de protección son la principal consideración, un desarrollo basado en transformaciones formales podría ser el más apropiado.

Análisis del riesgo

Se lleva a cabo el estudio de las causas de las posibles amenazas y probables eventos no deseados y los daños y consecuencias que éstas puedan producir. Se evalúan alternativas. Se debe tener un prototipo antes de comenzar a desarrollar y probar.

En resumen, es para tener en cuenta los riesgos de cada uno de los ámbitos.

Mecanismos de control

La dimensión radial mide el coste.

La dimensión angular mide el grado de avance del proyecto.

Variaciones del Modelo En Espiral.

Modelo en Espiral Típico de seis regiones

El modelo en espiral puede adaptarse y aplicarse a lo largo de la vida del software de computadora, a diferencia del modelo de proceso clásico que termina cuando se entrega el software.

Las 6 regiones que componen este modelo son las siguientes:

Comunicación con el cliente - Tareas necesarias para plantear la comunicación entre el desarrollador y el cliente.

Planificación - Tareas inherentes a la definición de recursos, el tiempo y otras informaciones relacionadas con el proyecto. Son todos los requerimientos.

Análisis de riesgos – Tareas para evaluar riesgos técnicos y otras informaciones relacionadas con el proyecto.

Ingeniería - Tareas para construir una o más representaciones de la aplicación.

Construcción y adaptación - Tareas requeridas para construir, probar, instalar y proporcionar soporte a los usuarios.

Evaluación del cliente - Tareas requeridas para obtener la reacción del cliente según la evaluación de las representaciones del software creadas durante la etapa de ingeniería e implementación durante la etapa de instalación.

Ventajas

El análisis del riesgo se hace de forma explícita y clara. Une los mejores elementos de los restantes modelos.

Reduce riesgos del proyecto

Incorpora objetivos de calidad

Integra el desarrollo con el mantenimiento, etc.

Además, es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático.

Desventajas

Genera mucho tiempo en el desarrollo del sistema

Modelo costoso

Requiere experiencia en la identificación de riesgos

Modelo de Cascada

Es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior. Al final de cada etapa, el modelo está diseñado para llevar a cabo una revisión final, que se encarga de determinar si el proyecto está listo para avanzar a la siguiente fase. Este modelo fue el primero en originarse y es la base de todos los demás modelos de ciclo de vida.

Fases:

Análisis de requisitos del software

En esta fase se analizan las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir. De esta fase surge una memoria llamada SRD (documento de especificación de requisitos), que contiene la especificación completa de lo que debe hacer el sistema sin entrar en detalles internos.

Es importante señalar que en esta etapa se debe consensuar todo lo que se requiere del sistema y será aquello lo que seguirá en las siguientes etapas, no pudiéndose requerir nuevos resultados a mitad del proceso de elaboración del software de una manera.

Diseño del sistema

Descompone y organiza el sistema en elementos que puedan elaborarse por separado, aprovechando las ventajas del desarrollo en equipo. Como resultado surge el SDD (Documento de Diseño del Software), que contiene la descripción de la estructura relacional global del sistema y la especificación de lo que debe hacer cada una de sus partes, así como la manera en que se combinan unas con otras.

Es conveniente distinguir entre diseño de alto nivel o arquitectónico y diseño detallado. El primero de ellos tiene como objetivo definir la estructura de la solución (una vez que la fase de análisis ha descrito el problema) identificando grandes módulos (conjuntos de funciones que van a estar asociadas) y sus relaciones. Con ello se define la arquitectura de la solución elegida. El segundo define los algoritmos empleados y la organización del código para comenzar la implementación.

Diseño del programa

Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario, así como también los análisis necesarios para saber qué herramientas usar en la etapa de Codificación.

Codificación del ordeñado

Es la fase en donde se implementa el código fuente, haciendo uso de prototipos, así como de pruebas y ensayos para corregir errores.

Dependiendo del lenguaje de programación y su versión se crean las bibliotecas y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido.

Pruebas

Los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente y que cumple con los requisitos, antes de ser entregado al usuario final.

Verificación

Es la fase en donde el usuario final ejecuta el sistema, para ello el o los programadores ya realizaron exhaustivas pruebas para comprobar que el sistema no falle.

Mantenimiento

Una de las etapas más críticas, ya que se destina un 75 % de los recursos, es el mantenimiento del software ya que al utilizarlo como usuario final puede ser que no cumpla con todas nuestras expectativas.

Ventajas

Realiza un buen funcionamiento en equipos débiles y productos maduros, por lo que se requiere de menos capital y herramientas para hacerlo funcionar de manera óptima.

Es un modelo fácil de implementar y entender.

Está orientado a documentos.

Es un modelo conocido y utilizado con frecuencia.

Promueve una metodología de trabajo efectiva: Definir antes que diseñar, diseñar antes que codificar.

Desventajas

En la vida real, un proyecto rara vez sigue una secuencia lineal, esto crea una mala implementación del modelo, lo cual hace que lo lleve al fracaso.

El proceso de creación del software tarda mucho tiempo ya que debe pasar por el proceso de prueba y hasta que el software no esté completo no se opera. Esto es la base para que funcione bien.

Cualquier error de diseño detectado en la etapa de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costos del desarrollo.

Una etapa determinada del proyecto no se puede llevar a cabo a menos de que se haya culminado la etapa anterior.

Modelo RUP

Es un proceso de ingeniería de software, que hace una propuesta orientada por disciplinas para lograr las tareas y responsabilidades de una organización que desarrolla software.

Su meta principal es asegurar la producción de software de alta calidad que cumpla con las necesidades de los usuarios, con una planeación y presupuesto predecible.

Características

Dirigido por Casos de Uso:

Los casos de uso son los artefactos primarios para establecer el comportamiento deseado del sistema

Centrado en la Arquitectura:

La arquitectura es utilizada para conceptualizar, construir, administrar y evolucionar el sistema en desarrollo

Iterativo e Incremental:

Maneja una serie de entregas ejecutables.

Integra continuamente la arquitectura para producir nuevas versiones mejoradas.

Conceptualmente amplio y diverso

Enfoque orientado a objetos

En evolución continua

Adaptable

Repetible

Permite mediciones:

Estimación de costos y tiempo, nivel de avance, etc.

Fases:**Inicio (Inception)**

El objetivo general de esta fase es establecer un acuerdo entre todos los interesados acerca de los objetivos del proyecto.

Es significativamente importante para el desarrollo de nuevo software, ya que se asegura de identificar los riesgos relacionados con el negocio y requerimientos.

Para proyectos de mejora de software existente, esta fase es más breve y se centra en asegurar la viabilidad de desarrollar el proyecto.

Elaboración

El objetivo en esta fase es establecer la arquitectura base del sistema para proveer bases estables para el esfuerzo de diseño e implementación en la siguiente fase.

La arquitectura debe abarcar todas las consideraciones de mayor importancia de los requerimientos y una evaluación del riesgo.

Construcción

El objetivo de la fase de construcción es clarificar los requerimientos faltantes y completar el desarrollo del sistema basados en la arquitectura base.

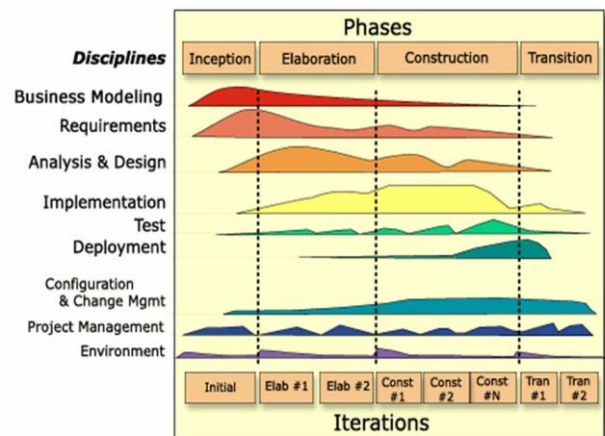
Vista de cierta forma esta fase es un proceso de manufactura, en el cual el énfasis se torna hacia la administración de recursos y control de las operaciones para optimizar costos, tiempo y calidad.

Transición

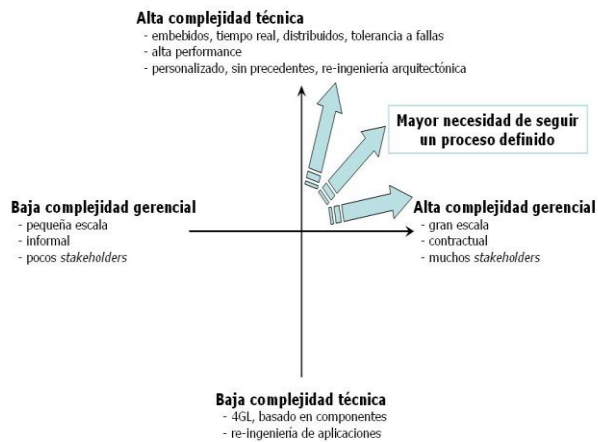
Esta fase se enfoca en asegurar que el software esté disponible para sus usuarios.

Se puede subdividir en varias iteraciones, además incluye pruebas del producto para poder hacer el entregable del mismo, así como realizar ajustes menores de acuerdo a ajuste menores propuestos por el usuario.

En este punto, la retroalimentación de los usuarios se centra en depurar el producto, configuraciones, instalación y aspectos sobre utilización.



¿Cuándo usar RUP?



Ventajas del Modelo RUP

Es el proceso de desarrollo más general de los existentes actualmente.

Es una forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo).

Desventajas del Modelo RUP

Método pesado

Por el grado de complejidad puede ser no muy adecuado.

En proyectos pequeños, es posible que no se puedan cubrir los costos de dedicación del equipo de profesionales necesarios.

10. Creé un infográfico para cada modelo de Desarrollo de software con las principales características.



CARACTERÍSTICAS MODELO ESPIRAL

CONTROL DEL RIESGO

Lo que diferencia en gran medida este modelo de los demás modelos de proceso de software es que reconoce explícitamente los riesgos. Por tanto, reduce considerablemente que fallen los proyectos grandes de software, ya que evalúa repetidamente los riesgos y verifica cada vez el producto en desarrollo.



GENÉRICO

Las cuatro etapas solo implantan los objetivos básicos de un ciclo, pero no tienen que manifestarse en cada ciclo. El orden de cada ciclo tampoco está estrictamente determinado. Por tanto, el modelo se puede combinar en cualquier momento con otros modelos.



METAMODELO

Incluye a los demás modelos. Por ejemplo, si la espiral fuera de un solo ciclo representaría al modelo de cascada, ya que incorpora el enfoque gradual de este modelo clásico. Utiliza el enfoque del modelo de creación de prototipos, ya que al comienzo de cada ciclo monta un prototipo para manejar los riesgos. Compatible con el modelo evolutivo, porque las iteraciones de la espiral se pueden considerar niveles evolutivos, a través de los cuales se construye el sistema final.

DESCRIPCIÓN DE LA ESPIRAL

Cada giro de la espiral representa un ciclo completo, por donde siempre pasan los cuatro cuadrantes, que representan las cuatro etapas del modelo. A medida que aumenta el tamaño de la espiral, también lo hace el progreso ejecutado. Por tanto, las etapas no se ejecutan solo una vez, sino varias veces, en forma de espiral.



FLEXIBLE

Es bastante flexible, al realizar por separado para cada fase del proyecto los procesos de definición de objetivos, análisis de riesgos, desarrollo y planificación.

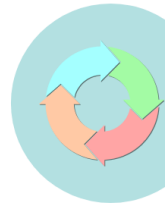


CARACTERÍSTICAS MODELO EN CASCADA



Este modelo fue el primero en originarse y es la base de todos los demás modelos de ciclo de vida.

Cada fase empieza cuando se ha terminado la fase anterior.



Para pasar de una fase a otra es necesario conseguir todos los objetivos de la etapa previa. Para ello se hace una revisión al final de la fase.

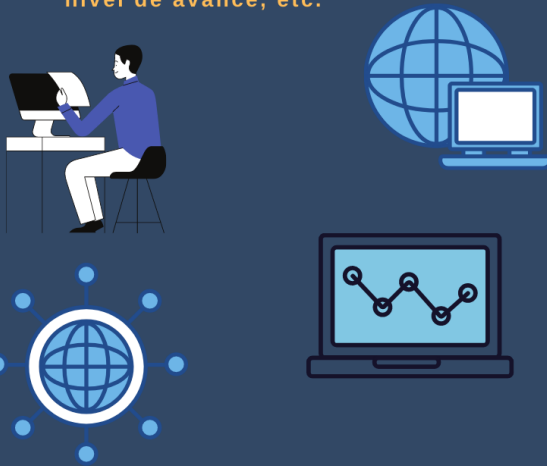
Ayuda a prevenir que se sobrepasen las fechas de entrega y costos esperados.



Al final de cada fase el personal técnico y los usuarios tiene la oportunidad de revisar el progreso del proyecto.

CARACTERISTICAS MODELO RUP

- **Dirigido por Casos de Uso:** Los casos de uso son los artefactos primarios para establecer el comportamiento deseado del sistema.
- **Centrado en la Arquitectura:** La arquitectura es utilizada para conceptualizar, construir, administrar y evolucionar el sistema en desarrollo.
- **Iterativo e Incremental:**
- **Maneja una serie de entregas ejecutables.**
- **Integra continuamente la arquitectura para producir nuevas versiones mejoradas.**
- **Conceptualmente amplio y diverso**
- **Enfoque orientado a objetos**
- **En evolución continua**
- **Adaptable**
- **Repetible**
- **Permite mediciones:**
- **Estimación de costos y tiempo, nivel de avance, etc.**



11. Realice el levantado de requisitos y comportamiento del sistema para un software en línea de alquiler de bicicletas, deberá contemplar el ejemplo visto en clase, para ello usted es el cliente, deberá llenar un documento con requisitos de hardware, conexión, y explicar el comportamiento del sistema, recuerde, en este caso usted es su propio cliente, pero deberá pensar en todos los datos que alimentan al sistema así como la forma en que sus clientes (Los que usen el sistema) entenderán el software.

Requisitos

- Software (alquiler de bicicletas)
- Hardware
- Servidor
- Estructura gerencial y administrativa.
- Sistema abonamiento, tarificación e identificación de personas usuarias.
- Distribución y localización de estaciones
- Estaciones: infraestructura, mobiliario urbano con módulos de estacionamiento, anclaje de bicicletas y de ubicación de equipos informáticos y de comunicaciones.
- Bicicletas: mecanismos electromecánicos de anclaje, identificación y lectura de bicicletas. Centro de control y monitoreo: dispositivos informáticos y de comunicación para gestionar el sistema y, sobre todo, para direccionar la logística de redistribución de bicicletas.
- Logística de redistribución: reposicionar o balancear el sistema por medio de la reubicación de bicicletas.
- Infraestructura y sistema de mantenimiento.
- Financiamiento.

Como cliente pediría un sistema si se puede usar la palabra llave en mano donde se contemplen los siguientes puntos:

Afiliación y Atención: Esto consta de un sistema de subscripción, tarificación e identificación de personas usuarias. Las herramientas tecnológicas se deben definir de acuerdo con el tipo de sistema que se va a implementar, que el sistema sea automatizado con el mayor acceso a toda la población.

Hardware y software, así como el mantenimiento y actualización

Sistema de información físico tótem o sistemas de información digital

Esquema de vinculación de personas usuarias al sistema

Sistema de información digital

Servicio al cliente

Accesibilidad para toda la población, brindando diversas opciones de operación del sistema

Marco legal**Mantenimiento y actualización.**

Se debe tener un centro de control, monitoreo y mantenimiento.

Espacio para vehículos de acarreo o servicio de bicicletas

- Área de desechos o residuos
- Taller de mantenimiento de bicicletas, terminales y anclajes
- Depósito de repuesto e insumos de bicicletas, terminales y anclajes
- Sala de control y equipos informáticos
- Oficinas administrativas
- Ubicación estratégica

Mantenimiento del sistema

- Inspección diaria del sistema
- Reporte diario de fallas
- Mantenimiento de estaciones y bicicletas
- Verificación y mantenimiento de las partes electrónicas.

Este mantenimiento del sistema es responsabilidad exclusiva a la empresa, la misma debe garantizar el funcionamiento de calidad y que el sistema funcione de manera adecuada.

Sistema de software de gestión para un sistema de alquiler de bicicletas.

El sistema de gestión de un Sistema de bicicletas es el conjunto de software de manejo de toda la información y datos que se requieren, se dividen en tres grandes procesos:

El sistema de identificación, validación, entrega y recibido de bicicletas: es un sistema para cada estación en donde sus funciones deben ser:

- Identificar una persona usuaria o una bicicleta.
- Validar la identidad de la persona usuaria (por consulta con la base de afiliados).
- Aceptar un nuevo usuario temporal (por código preestablecido o por solicitud en el terminal).
- Permitir el préstamo de la bicicleta.
- Recibir de regreso una bicicleta (después de su uso o inmediatamente en caso de avería)

El Sistema estará programado especialmente para operar e interactuar con lo siguiente:

- Equipo principal: terminal (lector de tarjeta, elementos, medios o sistemas de hardware pantalla, teclado).
- Medios de identificación y pago.
- Equipo de anclajes
- Bicicletas con identificación.
- Sistema de comunicación a distancia para consultas y validación con bases de afiliados.
- Permitir la integración con los otros componentes del software.
- Sistema general de control y comunicación central que a su vez está comunicado con el sistema de

identificación, información, afiliación y atención al usuario.

El sistema de información, afiliación y atención a personas usuarias: se requiere en la web para inscripción de personas usuarias y construcción de base de datos de identificación de usuarios y sus respectivas tarjetas.

Además, para facilitar la visualización de la misma podrá estar en móviles y tabletas.

El portal web debe contener:

Descripción del sistema: instrucciones generales de uso.

- Registro: instrucciones para el proceso de registro o abonamiento.

- Formas de pago: para el uso del sistema para no abonados.

- Formato de registro: para el registro de las membresías anuales, semestrales, mensuales, semanales e inscripción diaria.

- Pago en línea: a través de las mismas tarjetas de crédito aceptadas en las estaciones.

- Mapa de estaciones:

- Simbología de estaciones: debe mostrar rápidamente el estado y la disponibilidad de bicicletas.

- Contacto, debe contener detalles de los canales de atención al usuario disponibles (líneas telefónicas, centros de atención, dirección de correo electrónico, etc.).

El portal web se puede considerar completamente funcional si permite a usuarios y al público en general acceder a este y utilizar todas sus funciones con las más altas medidas de seguridad y una muy alta capacidad de consulta, incluso en los momentos de mayor carga o frecuentación.

El sistema general de control y comunicación: se requiere que sea instalado en un servidor del Centro de especializado. Permite la integración de todos los sistemas internos y externos y, sobre Control o en un hosting todo, la comunicación y validación entre las estaciones y el centro de control.

Brinda información en tiempo real, tanto en base de datos como de georreferenciación, sobre:

- Ubicación espacial de las estaciones e identificación de cada una de las bicicletas ancladas en estas.

- Disponibilidad de bicicletas y puntos de anclaje en cada una de las estaciones que indique:

Número de bicicleta y número de anclajes libres.

Señal de alarma de nivel de ocupación de la estación:

Si hay Saturación del servicio, 25 a 30% de anclajes están disponibles.

- Desocupada, 25 a 30% de los anclajes están ocupados.

- Relacionar un usuario específico a una única bicicleta.

- Comunicación de la terminal con el centro de control operacional, lo cual permitirá mantener en tiempo

real información sobre las transacciones realizadas en cada estación.

- Comunicación de los puntos de anclaje con la terminal de la estación.
- Comunicación de las bicicletas con los puntos de anclaje y con el centro de control operacional.
- Comunicación entre el módulo e interfaz con personal de la empresa concesionaria para la operación y planeación del Sistema (registros de GPS).
- Brinda datos del estado de la flota y estaciones a manera de inventario y disponibilidad.

Las funciones principales que debe cumplir este sistema son: recibir información de la base de datos y determinar si el usuario está bloqueado o no para acceder al préstamo, y así poder realizar las acciones correspondientes en la estación para liberar una bicicleta y controlar la interfaz del usuario en la estación, lo cual deberá permitir las siguientes acciones con las personas usuarias:

- Solicitar una bicicleta por usuarios con membresía.
- Compra de membresías a través de tarjeta bancaria.
- Verificar el estado de medio de pago especializado.
- Presentar contenidos del contrato de uso, información de la operación e información de contacto.
- Solicitar tiempo adicional en caso de que la estación se encuentre sin puntos de anclaje disponibles.