MANUAL TECNICO (Conceptualización y Ejecución)

MANUAL TECNICO

RETOMAMOS CON LO QUE ERA UN MANUAL TECNICO

- Habíamos mencionado que un manual técnico es un documento que contiene toda la información sobre los recursos utilizados por el proyecto, explicando todo el trabajo que se ha realizado al desarrollar un sistema de sw llevan una descripción muy bien detallada sobre las características físicas y técnicas de cada elemento.
- Y es importante porque va dirigido a la dirección de Tecnologia de información, al administrador del sistema y a otros desarrolladores de software para que puedan darle mantenimiento en caso que se requiera.
- Es por ello que decimos que un manual técnico de un determinado sistema es aquel que intenta dar asistencia a sus usuarios.

En otras palabras un manual tecnico Es un documento que contiene toda la informacion sobre los recursos utilizdos por el proyecto explicado todo el trabajo que se ha realizado al desarrollar un sistema o una actividad en concreto, lleva una descripcion muy bien detallada sobre las caracteriticas fisicas y tecnicas de cada elemento.

Por qué es importante el manual Técnico?

Porque encontramos la estructura de datos que usamos, cada función o procedimiento, cada variable, metodologías, etc. Esto sirve para futuras modificaciones, ya que es difícil recordar siempre la lógica con que se programa o puede que alguien más continué lo que empezaste y este documento es la mejor opción para retomar un proyecto o darle mantenimiento a nuestro sistema.

TIPOS DE MANUALES

Los manuales son textos utilizados como medio para coordinar y registrar información en forma sistémica y organizada. Un manual, además, es el conjunto de direcciones o instrucciones que se proponen guiar o mejorar la eficiencia de las tareas a realizar.

Existen distintos tipos de manuales:

1. **Manual organizacional**. Resume el manejo de una empresa en forma general. Indica la estructura, las funciones y roles que se cumplen en cada área. Es el más genérico y debe ser conocido por todos los empleadoss, ya que busca plasmar toda el funcionamiento de la institución, tanto la actividad administrativa, como la ejecutiva y operativa.
2. **Manual departamental**. Legisla y regula el modo en que deben ser llevadas a cabo todas las actividades de cada área. Las normas están dirigidas al personal según el departamento al que se pertenece y el rol que cumple. Estos manuales reflejan la forma de la organización pero a menor escala, es decir, regulan las actividades, las operaciones y los roles de cada departamento en particular.
3. **Manual de política**. Determina y regula la actuación y dirección de una empresa en particular. No tiene que ver con la política gubernamental (del país en el que funcione la empresa) si no con las políticas puntuales de la empresa. Este manual se enfoca en controlar y regular las actividades desempeñadas por los ejecutivos de la empresa.
4. **Manual de procedimientos**. Regula cada uno de los pasos que deben realizarse para emprender alguna actividad de manera correcta. Plantea las instrucciones para que los procesos (industriales, manufactureros y administrativos) que se llevan adelante en la organización, sean eficientes y eficaces.
5. **Manual de técnicas**. Explican minuciosamente cómo deben realizarse tareas particulares. Complementa al manual de procedimientos con indicaciones técnicas y detalladas sobre cada actividad.
6. **Manual de bienvenida**. Introduce brevemente la historia de la empresa, sus objetivos y su visión particular, además de los derechos, beneficios y obligaciones de los empleados. Busca generar la identificación de los nuevos ingresos con la empresa a la que están ingresando.
7. **Manual de puesto**. Determinan específicamente cuáles son las características y responsabilidades a las que se acceden en un puesto o rol determinado.

8. **Manual múltiple**. Expone distintas cuestiones, y suelen surgir de las minutos o de las actas de asambleas que se dan entre los socios. Estos manuales coordinan y organizan situaciones que no estén contempladas en el resto de los manuales de la organización.
9. **Manual de finanzas**. Verifica la administración de todos los bienes de la empresa, los ingresos y egresos. La creación y el seguimiento de este manual suele estar a cargo del tesorero y el controlador.
10. **Manual de sistema**. Se crea en el momento en que se desarrolla un sistema de gestión. Es frecuente que el departamento de tecnología sea el encargado de desarrollar este manual.
11. **Manual de calidad**. Presenta las políticas de la empresa en cuanto a la calidad del sistema. Puede estar relacionado con las actividades en forma sectorial o total de la organización. Presenta indicadores y parámetros que miden la productividad, la eficiencia y la calidad del servicio o producto que se ofrece.

ESTRUCTURA MANUAL TECNICO

- ✓ Portada (seguido de eso por lo general se debe tener el logotipo de la aplicación a desarrollar).

- ✓ Logotipo (ojo para este caso se lleva a cabo cuando ya se tenga la aplicación desarrollada, pero en este caso se elaborará por medio del desarrollo de prototipos hasta el momento).

- ✓ Contenido o Índice (es la Relación de los capítulos y páginas correspondientes que forman parte del documento).

- ✓ Introducción (Se debe presentar una breve descripción del sistema desarrollado, que contemple el ámbito abarcado, cual es su función principal y un detalle de las funciones macros o partes que lo componen. Puede incluir un mensaje de la máxima autoridad de las áreas comprendidas en el manual.)

- ✓ En la introducción describimos para que me va a servir este manual, de q trata y donde aplique esta actividad y se debe tener en cuenta el formato en que se encuentra realizado el documento ya que debe aplicar normas APA (tipo de letras, alineación, ortografía, gramática, etc) les doy un tips para corregir todo lo concerniente a gramática y ortigrafia en internet hay una pagina especializada en corregir este tipo de errores y se llama Spellboy, https://www.spellboy.com/corrector-gramatical/ en caso de q lo deseen utilizar les será de gran ayuda a nivel gramatical.

- ✓ Objetivos general y especifico (el general permiten conocer como su nombre lo dice de manera general en que consiste todo el documento) mientras los específicos (Se deben describir brevemente los objetivos específicos que se cumplieron con el desarrollo del sistema)

- ✓ Alcance (se especifica Que es lo que intentamos solucionar con la aplicación del sistema, y que es lo que en realidad solucionamos, ya que existen ocasiones en las que no se logran o por completo solucionar todo lo que aviamos requerido)

- ✓ Requerimientos técnicos (tenemos sw y hw)
  - Requerimientos sw
  - Requerimiento de hw

  (se socializa el sw que utilzairon para el desarrollo de su actividad principal ósea del proyecto describiéndolo paso a paso donde lo ideal que se cuente con imágenes del proceso realizado y como se llevaron a cabo sus actividades) y hw (se describe todo lo que se requirió o se necesitó para desarrollar el sw de ustedes ej: para el caso de requerimiento s de sw los proyectos que se enfocaron en desarrollo web o aplicación web entonces socializan el navegador que utilizaron, el sistema de gestor de datos, un emulador de un servidor eje SAMP).

- ✓ Software utilizado para el desarrollo
    - Ventajas del sw
- ✓ Descripción paso a paso de la actividad principal
- ✓ De contar con imágenes que ilustran el proceso
- ✓ Conclusión de sus actividades.

Lo principal de un MT es su introducción.

ENFATICEMOS EN TODO LO RELACIONADO CON contenido técnico muchachos PORQUE AQUÍ VAN

Requerimientos mínimos del hw (describe las características físicas que debe tener la computadora para el programa que van a desarrollar o van a usar, cuantos GB(yigabait) de RAM, Cuantos GB de disco duro, q procesador)

Requerimientos mínimos del sw (describe que SO utilice o estoy utilizando o caul planeo utilizar, si necesito permisos de administrador en el caso de q desee instalar un progama o si puedo trabajar en una cuenta de usuario normal o como invitado)

Requisitos mínimos de HW	Requisitos mínimos de SW
Cuanto necesito de Memoria RAM : 2 GB de RAM	Osea que Sistema operativo: windows 7 o superior. estoy utilizando
Cuanto necesito de Disco Duro 3 GB	Que tipo de Permisos: usuario/ administrador son los que voy a utilizar para este caso o este proyecto
Que Procesador necesito: 1,6 gigahertz (GHz) o superior	

Software utilizado para el desarrollo (VA que programa utilice para elaborar mi actividad o proyecto principal): ej elaboramos un proyecto enfocado con tablas dinámicas entonces un tip seria ir al navegador y averiguar cual seria para ese caso el sw utilizado describiendo sus caractriristicas y actividades ej: Microsoft Excel es una hoja de cálculo desarrollada por Microsoft para Windows, macOS, Android e iOS. Cuenta con cálculo, gráficas, tablas calculares y un lenguaje de programación macro llamado Visual Basic para aplicaciones

VENTAJAS DEL SW: con 5 ventajas como mínimo bien demarcadas se podría decir que estarían bien si desean mas no hay problema Aquí describimos porque use ese programa y no otro eje en el caso de las tablas dinámicas en Excel me voy al navegador y escribiría ventajas de Excel y solo extraería lo mas importante a nivel conceptual

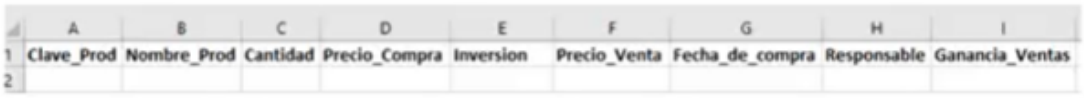
Entonces por lo general y para ahorrar tiempo lo único que hago es irme a mi navegador de mi preferencia que en este caso seria GOOGLE y le escribo en el buscador “requerimientos mínimos para instalar office 2016”

Ahora vamos a la descripción de paso a paso de la descripción principal

Descripción paso a paso de la actividad principal (aquí es donde se describen paso a paso como hacer su actividad principal, eje tablas dinámicas en excel) lo primero que debemos de tener es

Paso # 1 debemos de tener claros los pasos que debemos hacer en el caso de proyecto con tablas dinámicas seria

1. TENER REGISTRO EN UNA HOJA DE Excel los títulos de la tabla que se va a utilizar y estos deben estar marcados en negrita y anexo una pequeña captura de imagen



	A	B	C	D	E	F	G	H	I
1	Clave_Prod	Nombre_Prod	Cantidad	Precio_Compra	Inversion	Precio_Venta	Fecha_de_compra	Responsable	Ganancia_Ventas
2									

2. Registrar tabla con la información que será utilizada



	A	B	C	D	E	F	G	H	I
1	Clave_Prod	Nombre_Prod	Cantidad	Precio_Compra	Inversion	Precio_Venta	Fecha_de_compra	Responsable	Ganancia_Ventas
2	1	USB	165	$ 50.00	$ 8,250.00	$ 60.00	43931	JUAN	$ 9,900.00
3	2	MOUSE	29	$ 100.00	$ 2,900.00	$ 130.00	43949	FLOR	$ 3,770.00
4	3	TECLADO	108	$ 80.00	$ 8,640.00	$ 150.00	43951	LESLI	$ 16,200.00
5	4	AUDIFONOS	200	$ 20.00	$ 4,000.00	$ 25.00	43952	ASTRID	$ 5,000.00
6	5	CELULARES	20	$ 3,000.00	$60,000.00	$ 3,500.00	43935	RODOLFO	$ 70,000.00
7									

1. Índice

Relación de los capítulos y páginas correspondientes que forman parte del documento

2. Introducción.

Se debe presentar una breve descripción del sistema desarrollado, que contemple el ámbito abarcado, cual es su función principal y un detalle de las funciones macros o partes que lo componen. Puede incluir un mensaje de la máxima autoridad de las áreas comprendidas en el manual.

2.1. Objetivo general del sistema

Se debe de describir el objetivo general del sistema.

2.2. Objetivos específicos

Se deben describir brevemente los objetivos específicos que se cumplieron con el desarrollo del sistema.

3. Contenido técnico

3.1 DEFINICION DE REGLAS DEL NEGOCIO

Las **Reglas del Negocio** o Conjunto de **Reglas** de **Negocio** (Business Rules, por su descripción en inglés) describe las políticas, normas, operaciones, **definiciones** y restricciones presentes en una organización y **que son** de vital importancia para alcanzar los objetivos misionales.

3.2 **DIAGRAMA DE FLUJO DE DATOS JUNTO CON SU RESPECTIVO DICCIONARIO DE DATOS.**

Un **diagrama de flujo de datos** es una técnica muy apropiada para reflejar de una forma clara y precisa los **procesos** que conforman el sistema de información. Permite representar gráficamente los límites del sistema y la lógica de los **procesos**, estableciendo qué funciones hay que desarrollar.

Emplea símbolos definidos, como rectángulos, círculos y flechas, además de etiquetas de texto breves, para mostrar las entradas y salidas de **datos**, los puntos de almacenamiento y las rutas entre cada destino.

Un **diccionario de datos** es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

3.3 Descripción de campos requeridos por pantalla con presentación de pantallas.

Como su mismo nombre lo dice describe paso a paso la presentación del sw pero paso a paso y visaulizada de manera detallada, clara y fácil de interpretar

3.4. Diagrama de navegación del sistema.=

El **mapa o diagrama de navegación** es, básicamente, un gráfico o esquema en forma de árbol que representa la estructura o arquitectura general de un sistema. Es el primer paso para distribuir, organizar y jerarquizar el contenido que se verá en la pantalla de un producto o servicio digital.

Se les recomienda consultar los que cuentan con proyectos de aplicaciones web que los Diagramas de navegación en aplicaciones Web varían al contar con interacciones basadas en vínculos y lo pueden consultar con la siguiente palabra :**Diagramas de navegación en aplicaciones Web. Por ser** La web un sistema gráfico de información de hipertexto e hipermedia, distribuido, global, interactivo, dinámico e independiente de la plataforma, que funciona en Internet.

3.5. Requerimientos de interfase con otros sistemas

Aquí se definen las **interfaces** que soportará la aplicación. Debería contener adecuada especificidad, protocolos, puertos, direcciones lógicas, etc., tal que el software pueda ser desarrollado y verificado contra los estándares de requerimientos. Donde el objetivo es determinar las condiciones o capacidades que debe cumplir el sistema que se quiere diseñar para satisfacer las necesidades de un grupo de usuarios. Para lograr esto utilizaremos la definición de requerimientos. Un requerimiento se puede entender como una descripción informal de las necesidades y deseos que tiene el usuario final respecto a un producto de software.

Aquí debe ir

3.5. 1 interfaces de usuario (UI)= es el conjunto de los controles y canales sensoriales mediante los cuales un usuario puede comunicarse con una máquina. Por ejemplo, en una computadora, la pantalla, el teclado y las bocinas son parte de la interfaz de usuario porque la utilidad de todas ellas es hacer que entre o salga información del equipo.

3.5.2 interfaces de hw= Se trata de un conjunto de controles o dispositivos que permiten que el usuario intercambie datos con la máquina, ya sea introduciéndolos (pulsadores, botones, teclas, reguladores, palancas, manivelas, perillas) o leyéndolos (pantallas, diales, medidores, marcadores, instrumentos).

3.5.3 interfaces de sw= que se refieren al funcionamiento específico de los programas informáticos y de la información virtual que “ocurre” o “tiene lugar” dentro del computador.

Tal es el caso de las aplicaciones que empleamos a diario en nuestro trabajo con computadores. En otras palabras es todo lo visible en la pantalla del ordenador y la determinan los diseñadores. Cuenta con el diseño gráfico, los comandos, mensajes y demás elementos que determinan cómo interactuarán y navegarán los usuarios a través del contenido multimedia.

3.5.4 interfaces de comunicaciones= es la comunicación que establece la CPU del ordenador con cada uno de los dispositivos periféricos (teclado, ratón, teléfono móvil conectado a USB,) que se unen a ella a través de cables enlazados a sus respectivos puertos de conexión o conexiones inalámbricas con el fin de permitir el intercambio de información.

3.6. Modelo lógico de datos, diagrama entidad-relación.=

es un tipo de **diagrama** de flujo que ilustra cómo las "**entidades**", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema, se usan a menudo para diseñar o depurar bases de datos relacionales en los campos de ingeniería de software, sistemas de información empresarial, educación e investigación, además de que emplean un conjunto definido de símbolos, tales como rectángulos, diamantes, óvalos y líneas de conexión para representar la interconexión de entidades, relaciones y sus atributos.

3.7 Modelo de datos físico, junto con su respectivo diccionario de datos

es un **modelo** específico de bases de **datos** que representa objetos de **datos** relacionales (por ejemplo, tablas, columnas, claves principales y claves externas) y sus relaciones. ... Cree un **modelo físico** mediante la transformación de un **modelo de datos** lógicos.

El diccionario de datos es un listado organizado de todos los datos pertinentes al sistema, con definiciones precisas y rigurosas para que tanto el usuario como el analista tengan un entendimiento común de todas las entradas, salidas, componentes de los almacenes y cálculos intermedios. Contiene las características lógicas de los sitios donde se almacenan los datos

del sistema, incluyendo nombre, descripción, alias, contenido y organización. **También** identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información; se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.

3.8 PLATAFORMA DE USUARIO

Aquí se describen los requerimientos mínimos que se deben tener tanto de hardware como de software para que el sistema se pueda instalar y ejecutar correctamente.

Requerimientos mínimos de hw y sw

3.9 ÁREAS DE APLICACIÓN Y/O ALCANCE DE LOS PROCEDIMIENTOS. ESFERA DE ACCIÓN QUE CUBREN LOS PROCEDIMIENTOS. =

Aquí identifican los límites del proceso, su inicio, su finalización y su cobertura.**Mientras que en la Esfera de acción que cubren los procedimientos**. incluyen los criterios o lineamientos generales de **acción** que se determinan en forma explícita para facilitar la cobertura de responsabilidad de las distintas instancias que participaban en los **procedimientos**.

4. Responsables.

Para iniciar los trabajos que conducen a la integración de un manual, es indispensable prever que no queda diluida la responsabilidad de la conducción de las acciones en diversas personas, sino que debe designarse a un coordinador, auxiliado por un equipo técnico, al que se le debe encomendar la conducción del proyecto en sus fases de diseño, implantación y actualización. De esta manera se logra homogeneidad en el contenido y presentación de la información. Por lo que respecta a las características del equipo técnico, es conveniente que sea personal con un buen manejo de las relaciones humanas y que conozca a la organización en lo que concierne a sus objetivos, estructura, funciones y personal. Para este tipo de trabajo, una organización puede nombrar a la persona que tenga los conocimientos y la experiencia

necesarios para llevarlo a cabo. Por la naturaleza de sus funciones puede encargarlo al titular de el área específica. Asimismo, puede contratar los servicios de consultores externos.

4.1. Mapa de navegación.

muestra de forma gráfica la interconexión entre cada una de las pantallas del sistema, lo que serviría para saber como llegar a determinada parte de la aplicación. En este se muestran los menús, submenús y pantallas a las que nos lleva cada uno de ellos.

4.2. Descripción gráfica del mapa de navegación.

En el anterior aparece de forma de diagrama de flujo y en esta sección deberá aparecer ya con las respectivas pantallas.

Bibliografía

- ✓ A., A. (2021, 27 mayo). Concepto. MANUAL TÉCNICO. http://manualtcnico.blogspot.com/2011/08/concepto_21.html
- ✓ Sotomayor, D. (s. f.). ¿Qué es un manual técnico?, ¿Cuál es su estructura? Ejemplos. Scribd. Recuperado 27 de mayo de 2021, de https://es.scribd.com/document/361374373/Que-es-un-manual-tecnico-Cual-es-su-estructura-Ejemplos
- ✓ https://repositorio.unillanos.edu.co/bitstream/handle/001/323/Manual%20Tecnico%201.pdf?sequence=2
- ✓ https://www.youtube.com/watch?v=dIkUEBAHP2g
- ✓ https://es.calameo.com/read/004188934f6379e3c111a
- ✓ http://www.siiau.udg.mx/html/transparencia/appd/1.0.1.99/manual1_0_1_99.pdf.



Figura 1. Manual Técnico



Figura 2. Manual Técnico



Figura 3. Manual Técnico



Figura 4. Manual Técnico



Figura 5. Manual Técnico



Figura 6. Manual Técnico y Manual de Usuario

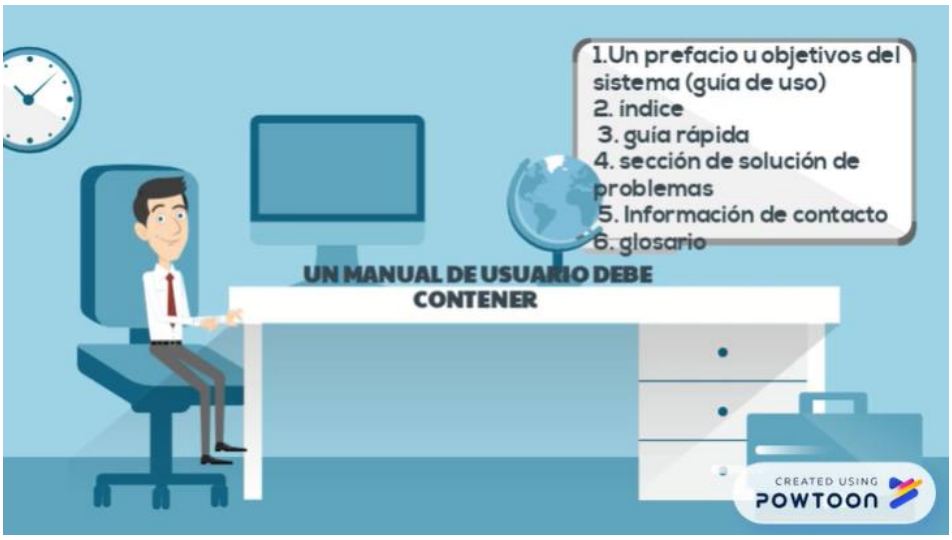


Figura 7. Manual Técnico y Manual de Usuario



Figura 8. Manual Técnico y Manual de Usuario



Figura 9. Manual Técnico y Manual de Usuario

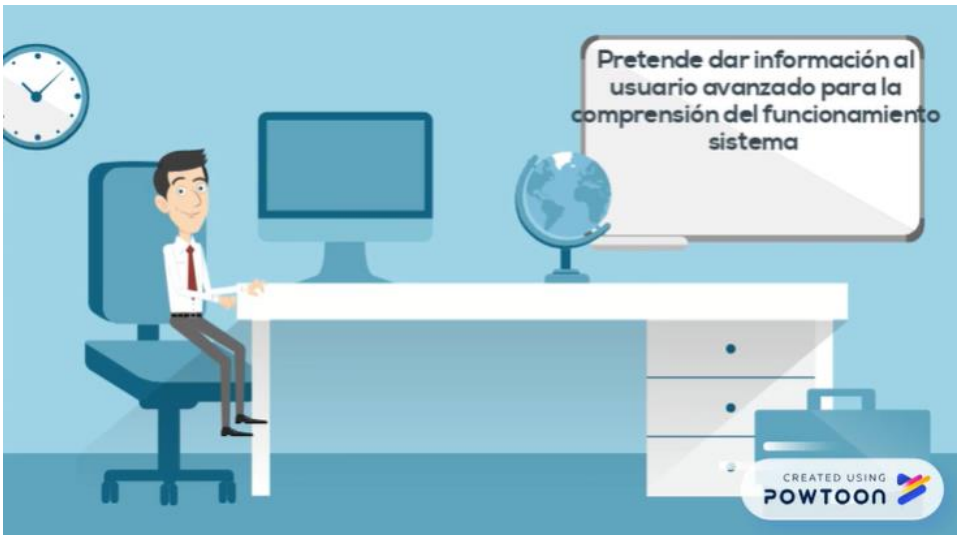


Figura 10. Manual Técnico y Manual de Usuario



Figura 10. Manual Técnico y Manual de Usuario



Figura 11. Manual Técnico y Manual de Usuario

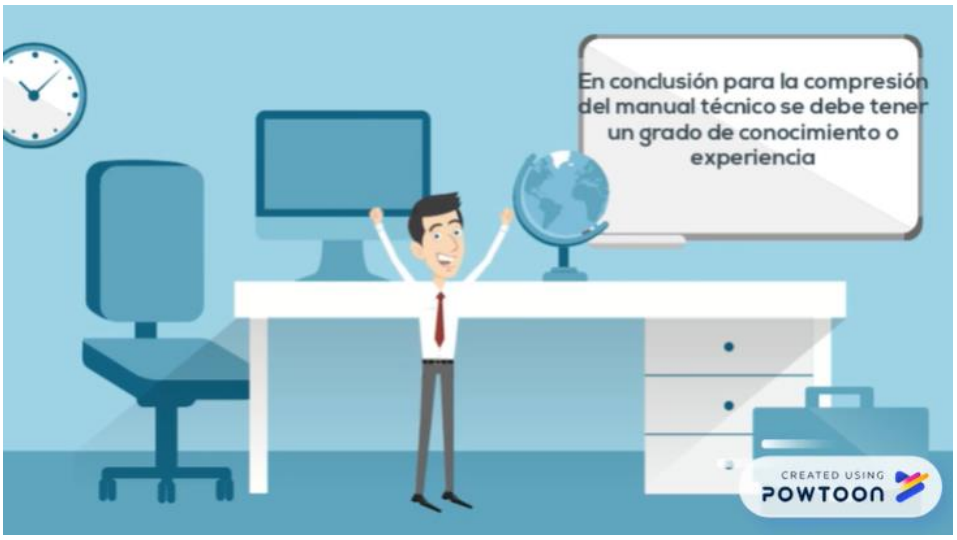


Figura 11. Manual Técnico y Manual de Usuario



Figura 12. Manual Técnico y Manual de Usuario

ARQUITECTURA DE SOFTWARE

La arquitectura de software de un programa ó de un sistema computacional es la estructura del sistema, que abarca componentes de software, las características externamente visibles de esos componentes, y las relaciones entre ellas.

AS es un nivel de diseño que considera "…no sólo algoritmos y estructuras de datos; el diseñar y especificar la estructura global del sistema emerge como un nuevo problema. Consideraciones estructurales tales como la organización y control global de la estructura; protocolos de comunicación y sincronización, acceso a datos; asignación de funcionalidad a los elementos del diseño; distribución física; composición de los elementos del diseño elements; crecimiento y desempeño; y selección de alternativas de diseño."

Identificando tres elementos principales: – procesamiento – datos – conección

Creación de una Arquitectura de Software

proceso DE as3

Un **proceso** es un conjunto de actividades planificadas que implican la participación de un número de personas y de recursos materiales coordinados para conseguir un objetivo previamente identificado.

Y En ingeniería, un proceso es una serie de tareas interrelacionadas que, juntas transforman las entradas en salidas.

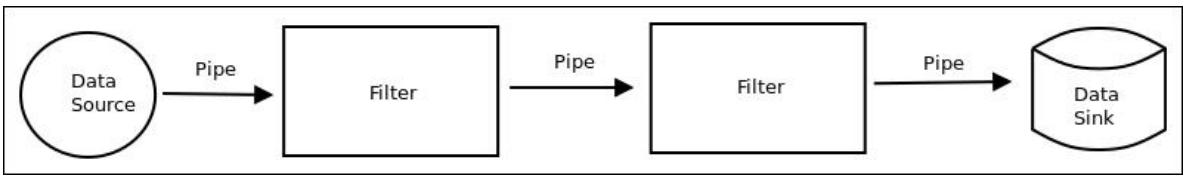
Ejemplos Arquitectura de software

Pipe and Filter ó PAIP AND FIOLTER (traduce **Tubería y filtro**) es un estilo arquitectónico simple que conecta varios componentes que procesan un flujo de datos, cada uno conectado al siguiente componente en la tubería de procesamiento a través de **Pipe** .

La arquitectura Pipe and Filter está inspirada en la técnica Unix de conectar la salida de una aplicación a la entrada de otra a través de tuberías en el shell.

La arquitectura de tuberías y filtros consta de una o más fuentes de datos. La fuente de datos está conectada a filtros de datos a través de tuberías. Los filtros procesan los datos que reciben y los pasan a otros filtros en la canalización.

Lo primero es entender el patrón Pipe and Filter. Este patrón se refiere a una cadena de **filtros** por los cuales pasa un mensaje (**Contexto**), desde el inicio al final. Cada uno de los filtros tiene acceso al contexto tanto de lectura como de escritura. Quien controla el armado y ejecución del Pipe es un **Controlador**. Este controlador lee de la configuración cuales son los filtros y su orden, los carga y le pasa el contexto al primero. Cuando el primero ejecuta le regresa el control al Controlador para que tome el Contexto y se lo pase al siguiente filtro.



data source = fuente de datos

PIPE = Tubería

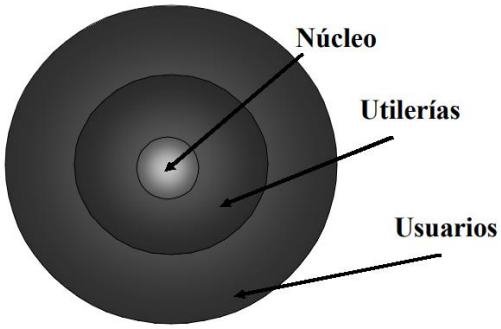
Filter = FILTRO

MANEJA =

- Ejecución concurrente (OSEA **concurrente** se utiliza para calificar a ALGO que se junta o SE UNE O incluso coincide con otro u otros aspectos en el mismo sitio o momento)
- Facilidad de Mantenimiento
- Soporte de Reutilización
- No soporta aplicaciones interactivas (Base de Datos)
- El Filtro más lento dicta la operación global del sistema

Organización por Capas

Capas

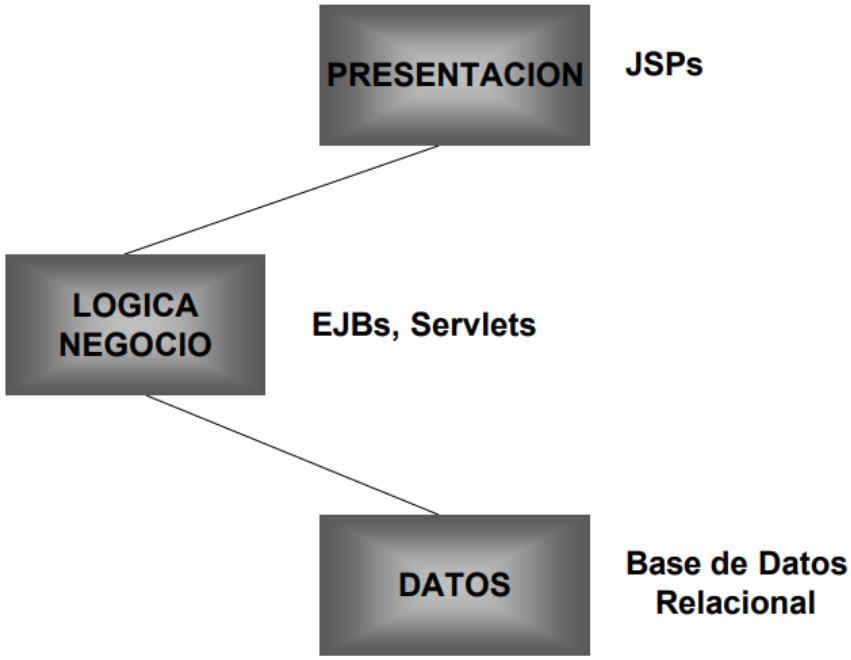


Manejan =

- Portabilidad
- Reduce complejidad del Sistema
- Fácil de Mantener
- No todos los sistemas se pueden representar por medio de capas.

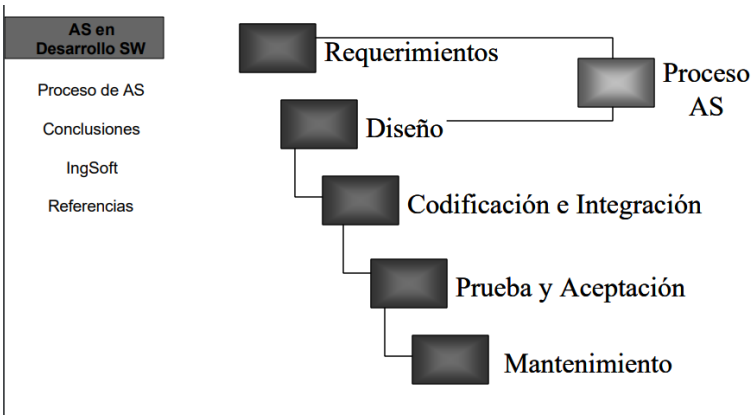
Arquitectura de 3 Niveles

JavaServer Pages (***JSP***) es una tecnología que ayuda a los desarrolladores de software a crear páginas web dinámicas basadas en HTML y XML



- Cada nivel puede correr en una maquina diferente ó en ambiente.
- multiprocesamiento, mejorando desempeño.
- Eficiente distribución de la aplicación.
- Promueve crecimiento.

DEFINCION DE EJBs = Los Enterprise JavaBeans (también conocidos por sus siglas EJB) son uno de las API que forman parte del estandar de construcción de aplicaciones empresariales J2EE de Sun Microsystems. Los EJBs proporcionan un modelo de componentes distribuido estándar para el lado del servidor. El objetivo de los Enterprise beans es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial (concurrencia, transacciones, persistencia, seguridad, ...) para centrarse en el desarrollo de la lógica de negocio en sí. El hecho de estar basado en componentes nos permite que éstos sean flexibles y sobre todo reutilizables.



- Medio de comunicación entre programadores, arquitectos, clientes usuarios, etc.
- Decisiones de desempeño, seguridad, mantenimiento, costos, etc.
- Promueve reutilización.

Impacto=

- Arquitectura del Sistema es la base en la toma de decisiones de diseño.
- La arquitectura deberá de satisfacer todos los requerimientos funcionales y no funcionales (calidad), así como anticipar los cambios de requerimientos y crecimientos.
- Detección de oportunidades de reutilización para: reducir costos (inversión, mantenimiento), mejorar calidad, mejorar tiempo de entrega (time-to-market).
- La Arquitectura del Sistema debe ser considerada en el mantenimiento para prevenir que haya violaciones en la arquitectura o malos entendidos de la misma.

Proceso de AS

definen “qué” hace y hará en el futuro la organización. La **arquitectura** de **procesos** es una colección de **procesos** de negocio y sus interdependencias entre ellos [6]; estos **procesos** deberían estar alineados a los objetivos de la organización.

SU Extracción se da a través de un Sistema Existente, luego de maneja o articula con un Sistema Nuevo se da inicio con – Análisis de requerimientos, considerando diferentes

perspectivas del sistema (cliente, usuario, programador, etc.) para Definir componentes. Además que

- ✓ Identificar componentes comunes en un dominio específico (familia de sistemas, líneas de productos)
- ✓ Definir una colección de componentes

Su reutilización se da a través de los siguientes parámetros: inicia con

Definir una librería de componentes reutilizables en el dominio y tiene en cuenta los

Nuevos sistemas utilizarán los componentes en la Librería



Es un proceso que involucra una metodología, principios, guías. 2. AS tiene como beneficios, reutilización, mejorar calidad, reducir costos, reducir tiempos de entrega (time-to-market). 3. AS en Dominios Específicos. 4. AS de Líneas de Productos

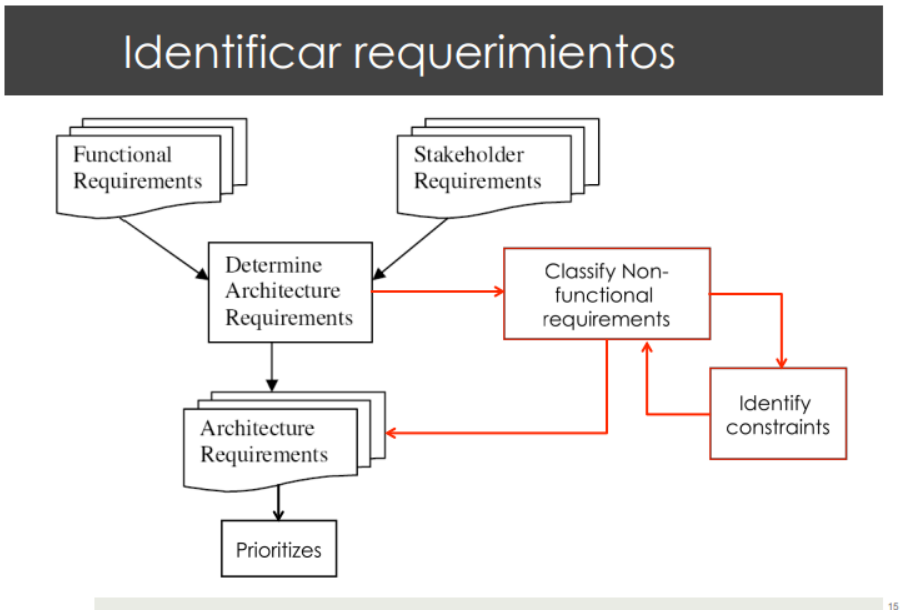
Etapas del Proceso de Arquitectura de software

Y vienen una serie de etapas definidas como Etapas del proceso de AS QUE CONSTAN DE

Definir los requerimientos: Involucra crear un modelo desde los requerimientos que guiarán el diseño de la arquitectura basado en los atributos de calidad esperados.

Diseño de la Arquitectura : Involucra definir la estructura y las responsabilidades de los componentes que comprenderán la Arquitectura de Software.

Validación: Significa "probar" la arquitectura, típicamente pasando a través del diseño contra los requerimientos actuales y cualquier posible requerimiento a futuro.



Stakeholders = partes interesadas

constraints = limitaciones.

Requerimientos no funcionales=

- ✓ Describen como el software debe comportarse, es decir como hacer algo, no que debe hacer.
- ✓ Están relacionados con los requerimientos funcionales porque describen la forma que se espera se logren dichos requerimientos.
- ✓ En algunos casos tienen restricciones de cómo hacerlo.
- ✓ Se clasifican de acuerdo al atributo de calidad esperado del sistema.

MIENTRAS QUE un requisito **funcional** define una función del sistema de software o sus componentes.

Las restricciones (constraints) imponen condiciones sobre la arquitectura que normalmente no son negociables. / Limitan el rango de alternativas de decisión del arquitecto / Algunas veces hace la vida más fácil para el arquitecto, en otras lo complica.

AHORA SI TENIENDO EN CEUNTA TODOS LOS CONCEPTOS DE AS PROCEDEMOS A LA CREACION DE LAS ETAPAS MAS IMPORTANTES

Requerimientos. La etapa de requerimientos se enfoca en la captura, documentación y priorización de requerimientos que influencian la arquitectura. Como se mencionó anteriormente, los atributos de calidad juegan un papel preponderante dentro de estos requerimientos, así que esta etapa hace énfasis en ellos. Otros requerimientos, sin embargo, son también relevantes para la arquitectura, estos son los requerimientos funcionales primarios y las restricciones.

Diseño. La etapa de diseño es la etapa central en relación con la arquitectura y probablemente la más compleja. Durante esta etapa se definen las estructuras que componen la arquitectura. La creación de estas estructuras se hace en base a patrones de diseño, tácticas de diseño y elecciones tecnológicas. El diseño que se realiza debe buscar ante todo satisfacer los requerimientos que influencian a la arquitectura, y no simplemente incorporar diversas tecnologías por que están "de moda".

Documentación. Una vez creado el diseño de la arquitectura, es necesario poder comunicarlo a otros involucrados dentro del desarrollo. La comunicación exitosa del diseño muchas veces depende de que dicho diseño sea documentado de forma apropiada. La documentación de una arquitectura involucra la representación de varias de sus estructuras que son representadas a través de distintas vistas. Una vista generalmente contiene un diagrama, además de información adicional, que apoya en la comprensión de dicho diagrama.

Evaluación. Dado que la arquitectura de software juega un papel crítico en el desarrollo, es conveniente evaluar el diseño una vez que este ha sido documentado con el fin de identificar posibles problemas y riesgos. La ventaja de evaluar el diseño es que es una actividad que se puede realizar de manera temprana (aún antes de codificar), y que el costo de corrección de los defectos identificados a través de la evaluación es mucho menor al costo que tendría el corregir estos defectos una vez que el sistema ha sido construido.

ROL del Arquitecto de software

Y ES AQUÍ DONDE EL ARQUITECTO DEL SW juega un papel de vital importancia para el desarrollo de sus proyecto ya que es un líder técnico que debe conocer los principios relacionados con la arquitectura de software, tener un amplio conocimiento respecto a la tecnología, y tener excelentes habilidades de comunicación escrita y oral.

Bibliografía

https://apiumhub.com/es/tech-blog-barcelona/principales-patrones-arquitectura-software/

https://es.wikipedia.org/wiki/Arquitectura_de_software

https://www.youtube.com/watch?v=5-l7p7apV3I&t=397s

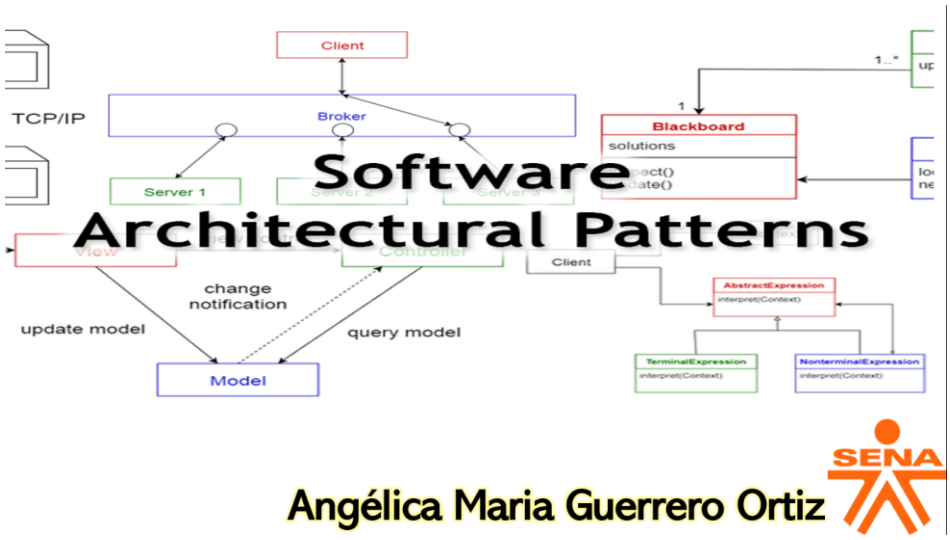


Figura 13. Arquitectura de Software



Figura 14. Arquitectura de Software

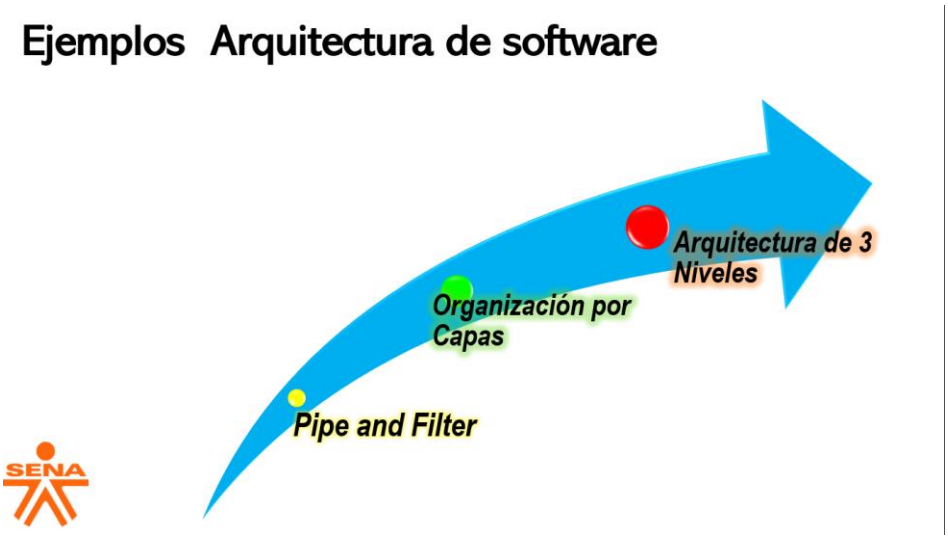


Figura 15. Arquitectura de Software

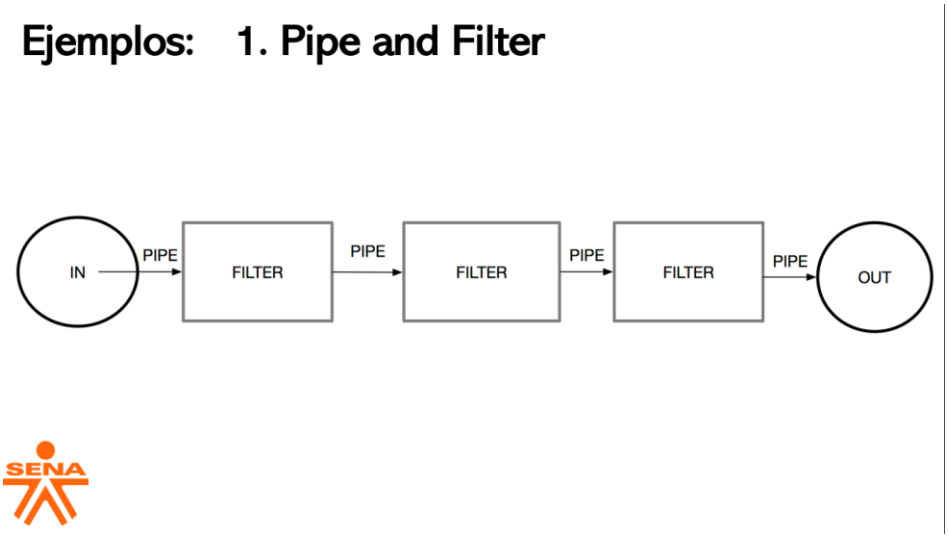


Figura 16. Arquitectura de Software



Figura 17. Arquitectura de Software

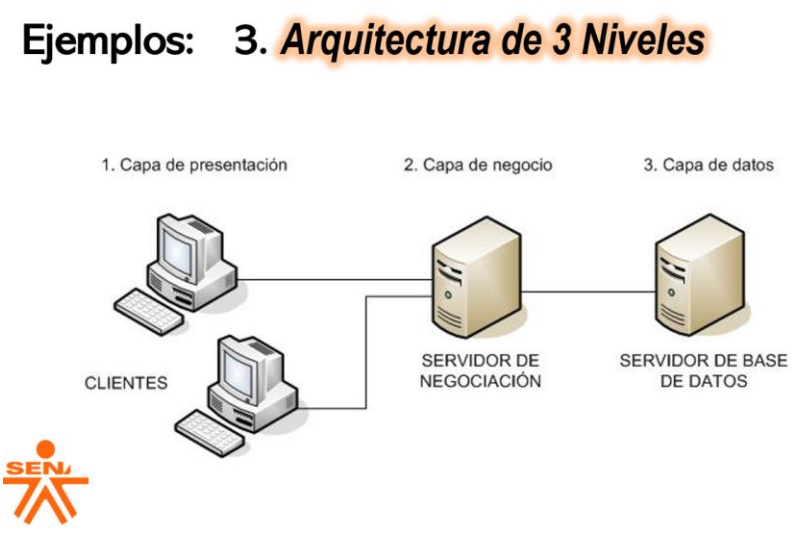


Figura 18. Arquitectura de Software



Figura 19. Arquitectura de Software



Figura 20. Arquitectura de Software



Figura 21. Arquitectura de Software

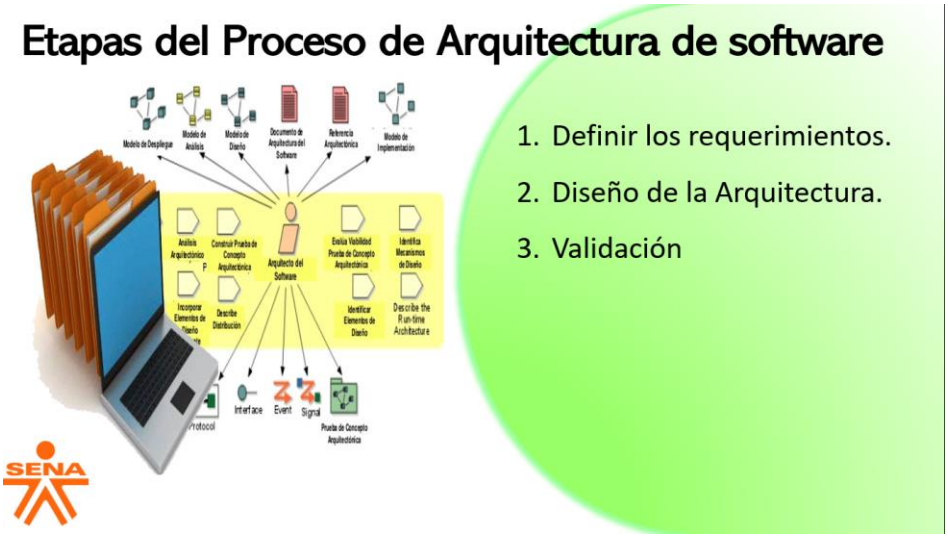


Figura 22. Arquitectura de Software

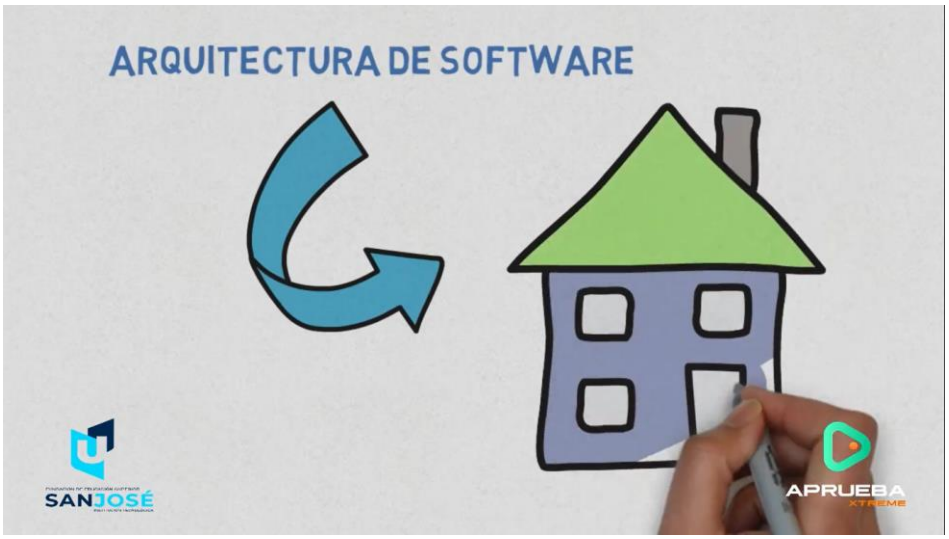


Figura 23. Arquitectura de Software



Figura 24. Arquitectura de Software

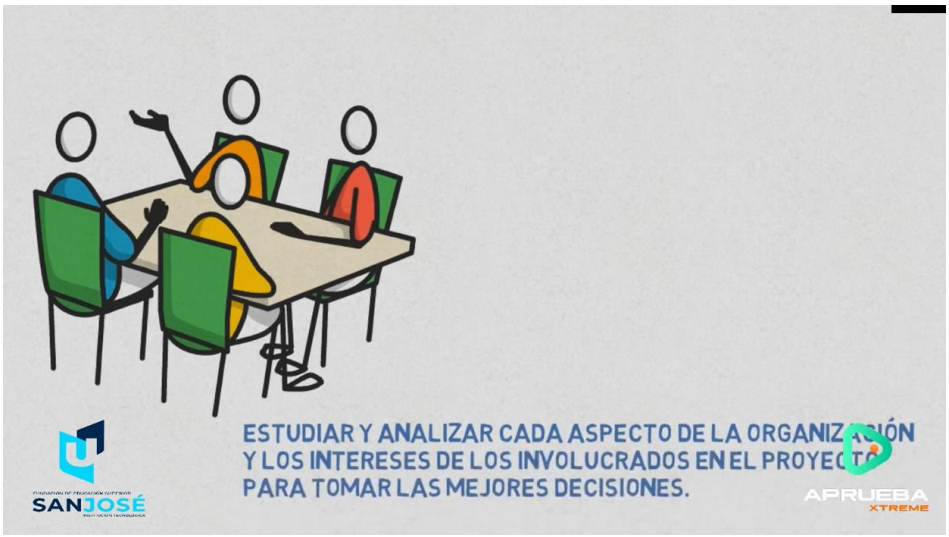


Figura 25. Arquitectura de Software



Figura 26. Arquitectura de Software



Figura 27. Arquitectura de Software



Figura 28. Arquitectura de Software

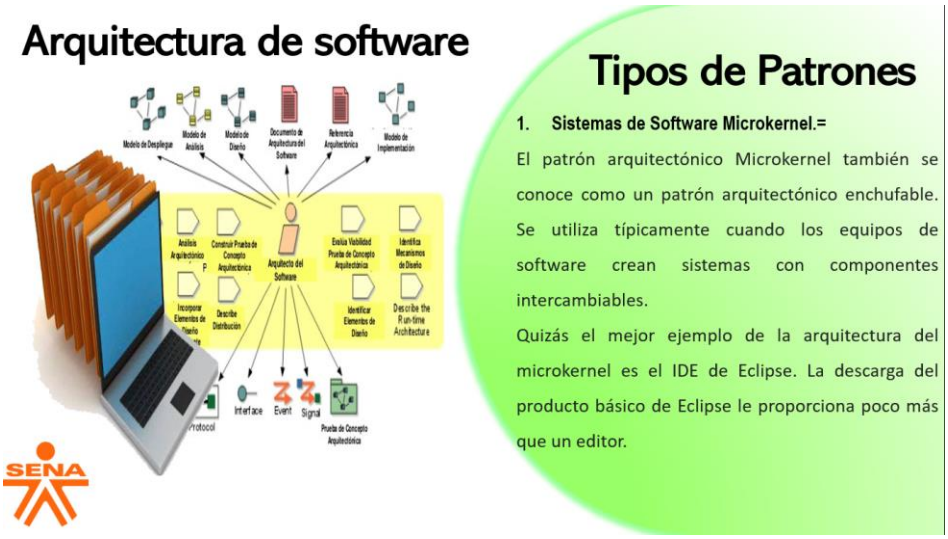


Figura 29. Arquitectura de Software



Figura 30. Arquitectura de Software



Figura 31. Arquitectura de Software



Figura 32. Arquitectura de Software



Figura 33. Arquitectura de Software



Figura 34. Arquitectura de Software



Figura 35. Arquitectura de Software



Figura 36. Arquitectura de Software

HISTORIAS DE USUARIOS

Que es una historia de usuario.

Una historia de usuario es una representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario.

Las historias de usuario son utilizadas en los métodos agiles para la especificación de requisitos, son una descripción breve de una funcionalidad software tal y como la percibe el usuario

Las historias de usuario son descripciones cortas y simples de una característica contada desde la perspectiva de la persona que desea la nueva capacidad, generalmente un usuario o cliente del sistema. Por lo general, siguen una plantilla simple:

Las historias de los usuario a menudo se escriben en fichas o notas adhesivas, se almacenan en una caja y se organizan en paredes o mesas para facilitar la planificación y el debate. Como tal, cambian fuertemente el enfoque de escribir sobre las características a discutir. De hecho, estas discusiones son más importantes que cualquier texto que se escriba.

Describen funcionalidades que dan solución a necesidades o problemas del cliente o del usuario, representan los “qués” a construir y se escriben en forma de historia con una o dos frases utilizando el lenguaje común del usuario.

Estas son una forma ágil de administrar los requisitos de los usuarios sin tener que elaborar gran cantidad de documentos formales los cuales sabemos que pocas personas lo llegan a leer completos y sin requerir de mucho tiempo para administrarlos debido ya que el mundo cambia muy rápido que por allí escribimos la necesidad y cuando se va a comenzar a desarrollar la misma cambio.

Porque Utilizar una Historia De Usuario

Utilizamos las historias de usuario porque siguen los principios básicos de requerimientos agiles

- ✓ Potencian la participación del equipo en la toma de decisiones.
- ✓ Se crean y evolucionan a medida que el proyecto avanza.
- ✓ Son peticiones concretas y pequeñas.
- ✓ Contiene la información imprescindible – Menos es más.
- ✓ Apoyan la cooperación, colaboración y conversación entre los miembros del equipo, lo que es fundamental.

¿Qué son?

Una manera simple de describir una tarea que aporta valor al usuario o al negocio. No se detalla mas hasta el momento que la historia de usuario se vaya a desarrollar.

Las HU pueden ser creadas durante las conversaciones con los usuarios interesados (stackholders) sobre nuevas funcionalidades o mejoras del proyecto.

Las HU es una invitación a la conversación.

Funcionalidades

Las historias de usuario forman parte de la formula de captura de funcionalidades definida en 2001 por Ron Jeffries de las tres C's:



podríamos decir que su comprensión es de fácil manejo pero el detalle aun no está definido y esto se tendrá en cuenta con la planificación del SPRINTF es por esta razón que se debe tener muy claro lo que realmente se requiere en un sw a desarrollar de lo contrario la HU será rechazada o caso contrario será construida de forma incorrecta y para que esto no pase debemos tener en cuneta que las HU deben cumplir con las 3 C

1. CARDS o tarjeta= significa que deben estar escritas como su nombre lo dice en una tarjeta, en lenguaje de negocio que sirve como identificación y recordatorio del requerimiento y ayuda para la planificación mediante la priorización. tarjeta refleja los elementos más importantes de la historia de usuario. Expresa el valor que se quiere conseguir desde el punto de vista del usuario.

2. CONVERSACION= es donde van los detalles de las HU las cuales deben ser hablados o conversados. Aquí el dialogo que ocurre entre los miembros del equipo y el dueño del podricto o el (producto owner) es para aclarar los detalles y dudas que surgan sobre las HU, y es la parte mas importante de las historias que ustedes van a desarrollar, por eso, Después de escribir la historia de usuario en la tarjeta, es necesario tener una conversación al respecto de su contenido. Hay que saber responder a cuestiones sobre el valor y sobre el resultado esperado de la implementación. Esta conversación puede suceder en cualquier momento; es habitual que se produzca durante el refinamiento del Backlog o el Sprint Planning.

CONCFIRMACION = es donde el dueño del producto (producto owner) debe confirmar que el equipo comprenda y entienda la HU no solo contarla sino también explicarla, además de que las pruebas se llevaran a cabo para poder decir si la HU se ha completado con exito. En otras palabras La confirmación es un acuerdo que refleja que todas las personas implicadas, Product Owner y Development Team (Equipo de desarrollo), entienden cuales son los elementos, valor y resultado esperado de la historia de usuario. Muchos equipos cometen el error de no tener esta confirmación, lo cual lleva a fricciones y problemas durante el Sprint Planning (planificador)

Para poder entender correctamente la estructura y diseño de una HU lo vamos a realizar inicialmente de manera manual antes de irnos a trabajarla de manera digital.

En el mundo real debemos buscar herramientas o técnicas para poder comunicar de manera solida y concisa lo que requerimos, hoy en día en los ambientes agiles la herramienta mas popular para llevar a cabo este proceso son las famosas HU, pero para poder crearlas se requieren de 3 pasos necesarios para generar una buena historia de usuario

1. El primer componente para crear una HU, es la carta o también conocida como una famosa promesa de discusión y de vital importancia porque muchas empresas piensan que con que se termine la estructura hasta este momento ya se formaría parte de una historia y no es así, porque en la primera parte donde escribimos la carta de la siguiente manera

Donde USER Persona o usuario quiero cierta funcionalidad para obtener algún beneficio eje como usuario de mi celular ANDROID quiero establecer una nueva alarma donde se visualice el tiempo que falta para que esta se active para estar seguro que la establecí correctamente o para saber cuanto tiempo voy a dormir, con este ejemplo solo tendríamos parte de la carta de la historia escrita en el formato recomendado.

2. En la siguiente parte o 2 componente viene lo referente a la descripción y se lleva a cabo en la junta de refinamiento con el equipo y se describe la funcionalidad que se espera obtener al final de la historia

3 Por ultimo y como 3 paso de la creación de la historia o también conocido como 3er componente tenemos la confirmación dentro de ella se definen los criterios de aceptación que también son muy famoso o muy conocidos y dentro de estos de estos se informa que condiciones deben cumplirse para q la historia se de por terminada, delimita el SCOPE de la historia y valida las expectativa del trabajo a realizar y volviendo al ejemplo de la alrma del celular entonces seria el de la aclaración de algunos puntos como el texto blanco

con fondo gris o texto gris con fondo blanco dependiendo del tema que este activo en el celular, así como de las características que deben cumplir con la funcionalidad para saber que la historia ya se termino

¿CUÁNDO SE ESCRIBEN LAS HISTORIAS DE USUARIO?

Las historias de usuario se escriben en todo el proyecto ágil. Por lo general, se lleva a cabo un taller de redacción de historias de usuario cerca del inicio del proyecto ágil. Todos en el equipo participan con el objetivo de crear un Product Backlog (Pila de Producto) que describa por completo la funcionalidad que se agregará durante el transcurso del proyecto o un ciclo de lanzamiento de tres a seis meses.

Algunas de estas historias de usuario ágiles serán, sin duda, épicas. Las épicas se descompondrán más tarde en historias más pequeñas que caben más fácilmente en una sola iteración. Además, las historias nuevas se pueden escribir y agregar al Product Babklog en cualquier momento y por cualquier persona.

Información en una historia de usuario

Para decidir qué información incluir en una historia de usuario es preferible no adoptar formatos rígidos. Los resultados de scrum y agilidad no dependen de las formas, sino de la institucionalización de sus principios y la implementación adecuada a las características de la empresa y del proyecto.
Los campos que se consideran más necesarios para describir de una manera adecuada las historias de usuario son:

Por lo anterior debemos saber que las historias de usuario son de cada cliente y deben estar enfocadas a un entregable que pueden ser usados por el usuario final, entonces recordemos toda historia de usuario debe tener 3 aspectos importantes

1. Descripción descripción sintetizada de la historia de usuario. El estilo puede ser libre, según mejor nos funcione, debe responder a tres preguntas: ¿Quien se beneficia? ¿Qué se quiere? y ¿Cuál es el beneficio? Mike Cohn recomienda seguir el siguiente patrón

que garantiza que la funcionalidad está descrita a un alto nivel y de una manera no demasiado extensa:

Como [rol del usuario], quiero [objetivo], para poder [beneficio]

"El valor de cada historia esta en el beneficio por lo cual debería ser lo mas importante, por eso lo colocamos al inicio", desde allí comencé a escribirlas de esta manera, pero cada uno de nosotros las puede escribir como quiera

2. Estimación estimación del esfuerzo necesario en tiempo ideal de implementación de la historia de usuario. Según convenga al equipo también se puede utilizar unidades de desarrollo, conocidas como puntos de historia (estas unidades representan el tiempo teórico de desarrollo/persona que se estipule al comienzo del proyecto).

3. Priorización= sistema de priorización que nos permite determinar el orden en el que las historias de usuario deben de ser implementadas.

Esos son los campos mínimos que debe incluir una historia de usuario, también pueden incluir ID, Titulo, Criterios de Aceptación, Valor, Dependencias, Persona Asignada, Sprint, Riesgo, Modulo y observaciones entre otros.

Criterios de Aceptación y se recomienda tener en cuenta agregar los criterios de validación o aceptación que sirven para definir si una HU esta bien construida ahora vamos a ver como se crea una HS

La descripción se escribe así= Yo, como roll y aquí en el roll se escribe a quin esta dirigida la HU y quien la utilziara por ej= YO como cliente pero un error típico seria el de registrar yo como banco ya que no utilizaran esa funcionalidad y lo mas probable es que sea un objetivo del negocio.

Luego viene el quiero, aquí esta la necesidad a cubrir eje quiero ingresar productos al sistema

Y finalmente el para, donde esat el objetivo a lograr por ej para vender mis productos por la web.

Con lo anteriormente dicho la HU quedaría de la siguiente manera

Yo, como….. cliente Quiero……. Ingresar mis productos al sistema Para…….. venderlos por la web

Con este ejemplo podríamos decir que aplicaríamos la reglas de las HU con las 3C

Para poder entender correctamente la estructura y diseño de una HU lo vamos a realizar inicialmente de manera manual antes de irnos a trabajarla de manera digital.

- Como <Usuario>
- Quiero <algún objetivo>
- Para que <motivo>

Ejemplos de historias de usuario (aplicables a SCRUM/KANBAN):

COMO usuario QUIERO poder buscar transacciones PARA poder detectar gastos innecesarios en mi cuenta durante cierto periodo de tiempo

Ejemplo Historia de Usuario 1 – Transacciones

- ROL: Usuario
- OBJETIVO: Poder buscar transacciones
- MOTIVACIÓN: Detectar gastos innecesarios en mi cuenta durante cierto tiempo

COMO usuario QUIERO poder acceder a la agenda de enfermería PARA utilizar sus funcionalidades

Ejemplo Historia Usuarios 2 – Agenda

- ROL: Usuario
- OBJETIVO: Poder acceder a la agenda
- MOTIVACIÓN: Utilizar funcionalidades

El título de una historia de usuario debe mapear una funcionalidad de nuestro producto o software.

Cuando escribimos historias de usuario, es una buena práctica el usar personajes en vez de roles, ya que dan incluso más contexto. Por ejemplo:

COMO usuario QUIERO ser capaz de mostrar la lista de visitas PARA poder gestionar la llegada de pacientes y de salas de espera
vs:

COMO enfermera QUIERO ser capaz de mostrar la lista de visitas PARA poder gestionar la llegada de pacientes y de salas de espera
Con el simple cambio de "usuario" a "enfermera" estamos dando mucho más contexto a nuestra historia de usuario y la hacemos mucho más legible para cualquier persona sin background técnico.

Debemos recordar que el título de una historia de usuario debe mapear una sola funcionalidad de nuestro producto o software.

Como <Usuario> para quién desarrollamos

Quiero <algún objetivo> aquí describimos su intención, no las funciones que usan.

Para que <motivo> cómo encaja su deseo inmediato de hacer algo en la perspectiva general

COMO usuario QUIERO poder buscar transacciones PARA poder detectar gastos innecesarios en mi cuenta durante cierto periodo de tiempo

Ejemplo Historia de Usuario 1 – Transacciones

ROL: Usuario

OBJETIVO: Poder buscar transacciones

MOTIVACIÓN: Detectar gastos innecesarios en mi cuenta durante cierto tiempo

Beneficios

Para los equipos de desarrollo nuevos en la metodología ágil, las historias de usuario a veces parecen un paso más. ¿Por qué no dividir el gran proyecto (el epic) en una serie de pasos y seguir adelante? Pero las historias dan al equipo un contexto importante y asocian las tareas con el valor que estas aportan.
Las historias de usuario tienen varios beneficios clave:

- Las historias centran la atención en el usuario. Una lista de tareas pendientes mantiene al equipo centrado en tareas que deben completarse, pero un conjunto de historias lo mantiene centrado en solucionar problemas para usuarios reales.
- Las historias permiten la colaboración. Con el objetivo definido, el equipo puede colaborar para decidir cómo ofrecer un mejor servicio al usuario y cumplir con dicho objetivo.
- Las historias impulsan soluciones creativas. Las historias fomentan que el equipo piense de forma crítica y creativa sobre cómo lograr mejor un objetivo.
- Las historias motivan. Con cada historia el equipo de desarrollo disfruta de un pequeño reto y una pequeña victoria, lo que aumenta la motivación.

CARACTERISTICAS

- Independientes unas de otras: De ser necesario, combinar las historias dependientes o buscar otra forma de dividir las historias de manera que resulten independientes.
- Negociables: La historia en sí misma no es lo suficientemente explícita como para considerarse un contrato
- Valoradas por los clientes o usuarios: Los intereses de los clientes y de los usuarios no siempre coinciden, pero en todo caso, cada historia debe ser importante para alguno de ellos más que para el desarrollador.
- Estimables: Un resultado de la discusión de una historia de usuario es la estimación del tiempo que tomará completarla.

- Pequeñas: Las historias muy largas son difíciles de estimar e imponen restricciones sobre la planificación de un desarrollo iterativo
- Verificables: Las historias de usuario cubren requerimientos funcionales, por lo que generalmente son verificables

Ventajas que aportan las historias de usuario:

- ✓ Al ser muy cortas, estas representan requisitos del modelo de negocio que pueden implementarse rápidamente (días o semanas).
- ✓ Necesitan poco mantenimiento.
- ✓ Mantienen una relación cercana con el cliente.
- ✓ Permiten dividir los proyectos en pequeñas entregas.
- ✓ Permiten estimar fácilmente el esfuerzo de desarrollo.
- ✓ Son ideales para proyectos con requisitos volátiles o no muy claros.

Desventajas de las historias de usuario

No todo son ventajas, aunque si es verdad, que los pocos inconvenientes que tienen estas prácticas, son fácilmente solucionables.

- Es difícil de medir la eficacia de un proyecto con los resultados de una historia, aunque estos hayan sido positivos, pues necesitaremos una visión más global para poder clasificar un proyecto terminado como correcto.
- Necesitaremos desarrolladores cualificados que tengan una visión muy clara sobre sus posibilidades y cómo conseguir objetivos a corto plazo, puesto que las historias se desglosan en pequeñas tareas.

- Para poder asumir la escalabilidad de un proyecto grande, necesitaremos de profesionales, pues las historias nos dan pistas, pero es necesario tener visión a largo plazo para agrupar las historias y ofrecer datos más precisos.
- Es necesario un continuo feedback con el cliente para resolver posibles dudas o incluso para ir agregando historias a nuestros sprints.

ESTRUCTURA

1. Comenzamos con ponerle un nombre a la HU y al no ponerle un nombre, lo más seguro es que durante el proceso de desarrollo de sus proyectos el caos se hará presente ¿por qué? Porque luego no sabe realmente con que HU están trabajando finalmente o incluso pueden llegar a perderse, por lo anterior es importante, poner nombre a su HU y la recomendación más importante aquí es comenzar por verbos.

2. Después del nombre vamos con la Descripción de la HU debe ser muy común que generalmente comienza como <rol> y cuando hablamos del rol hablamos específicamente de la persona que va a desempeñarse en esta HU porque vamos a crear una historia, por ej : como empleado, como cajero, como automovilista, ESTABLECER ese rol y dejarlo bien claro pero el error más común es cuando se registra como USURIO esto lo utilizan mayormente en las HU y utilizan mucho la palabra usuario pero debemos tener muy claro y debemos establecer quien va a ser esa persona que v a formar la parte de esta historia .

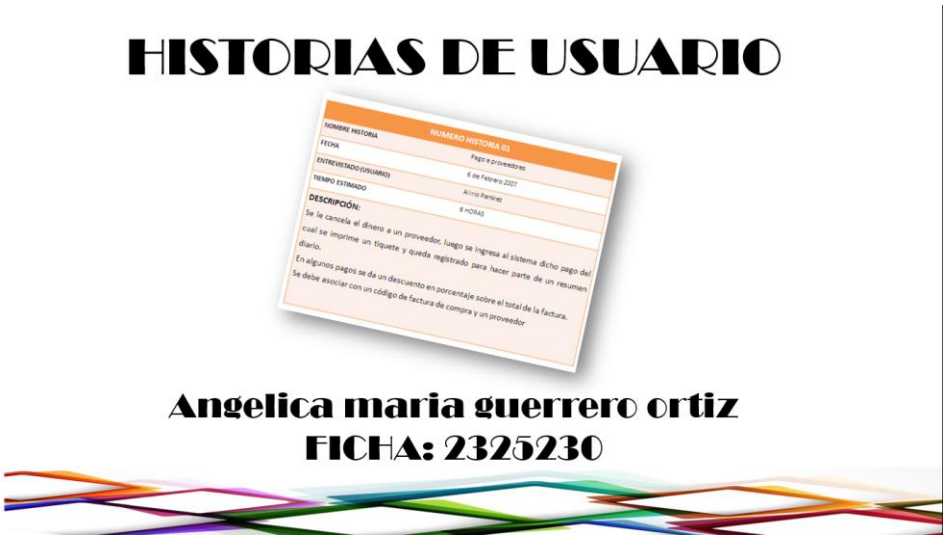


Figura 37. Historia de Usuario



Figura 38. Historia de Usuario



Figura 39. Historia de Usuario

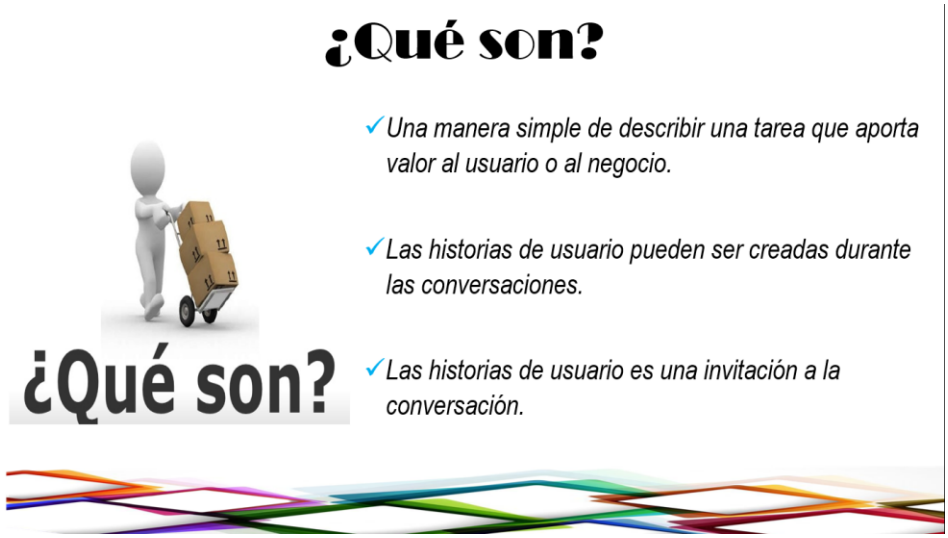


Figura 40. Historia de Usuario



Figura 41. Historia de Usuario



Figura 42. Historia de Usuario

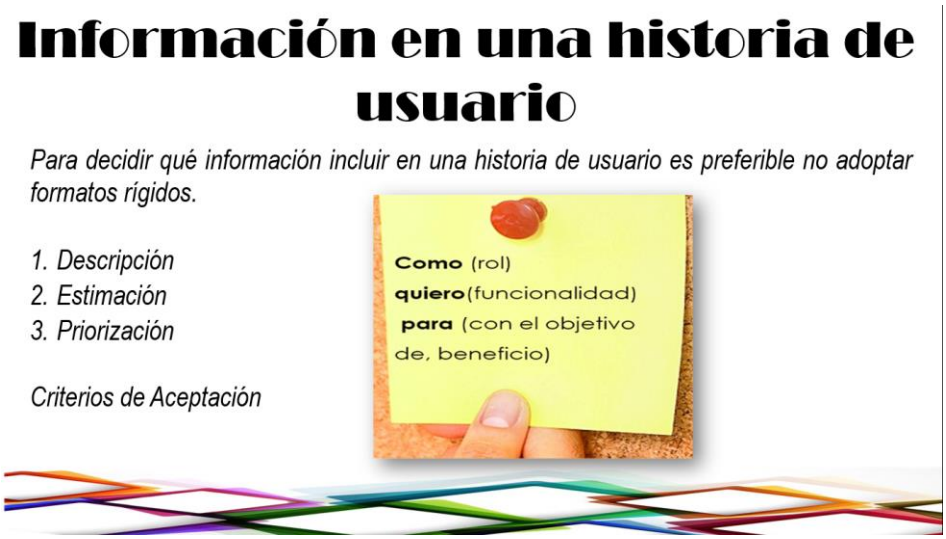


Figura 43. Historia de Usuario

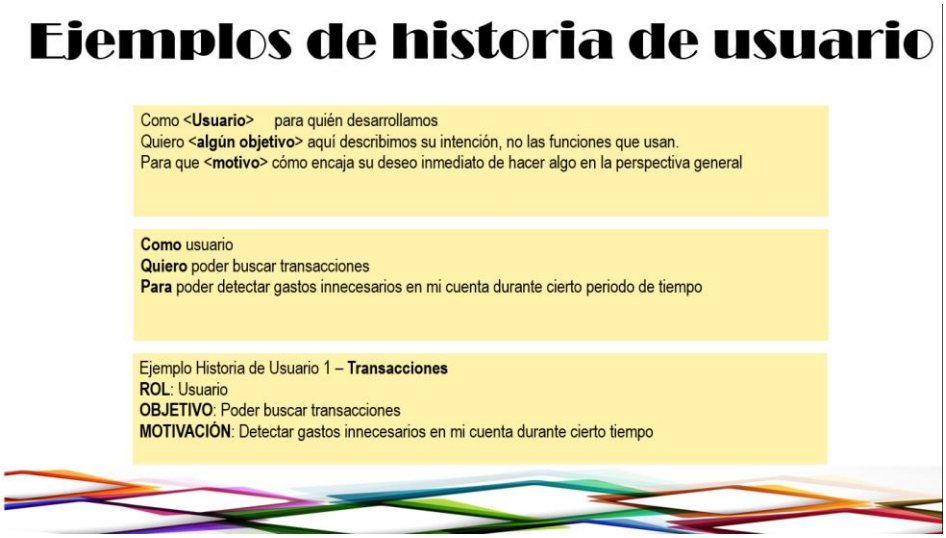


Figura 44. Historia de Usuario

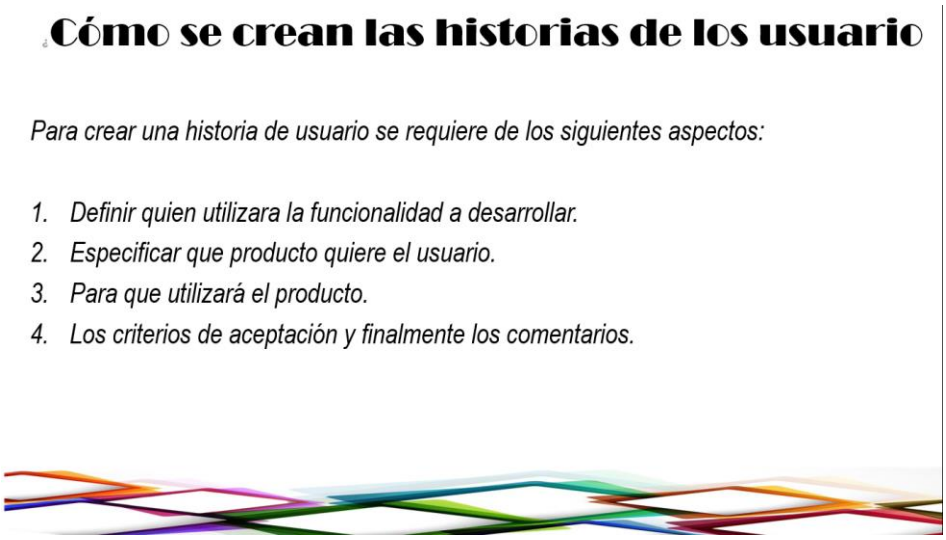


Figura 45. Historia de Usuario



Figura 46. Historia de Usuario



Figura 47. Historia de Usuario



Figura 48. Historia de Usuario

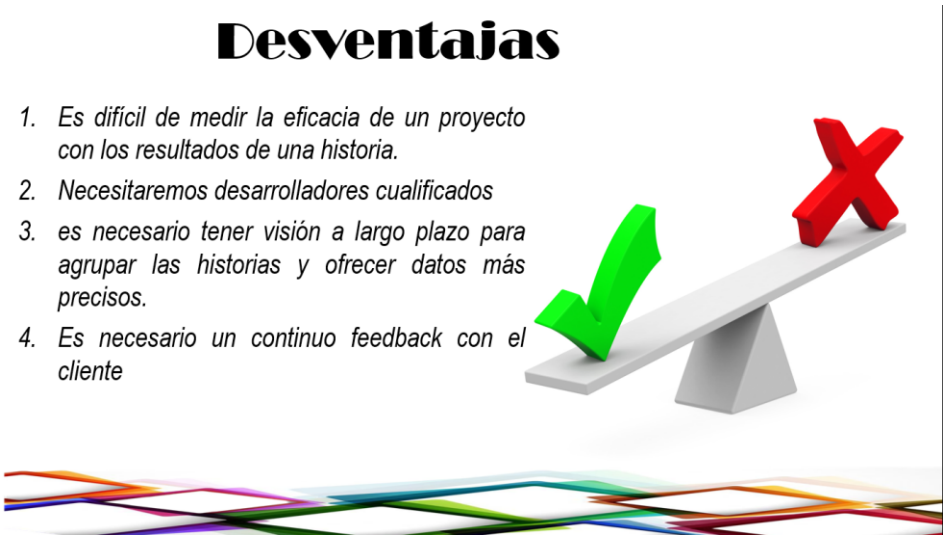


Figura 49. Historia de Usuario



Figura 50. Historia de Usuario



Figura 51. Historia de Usuario

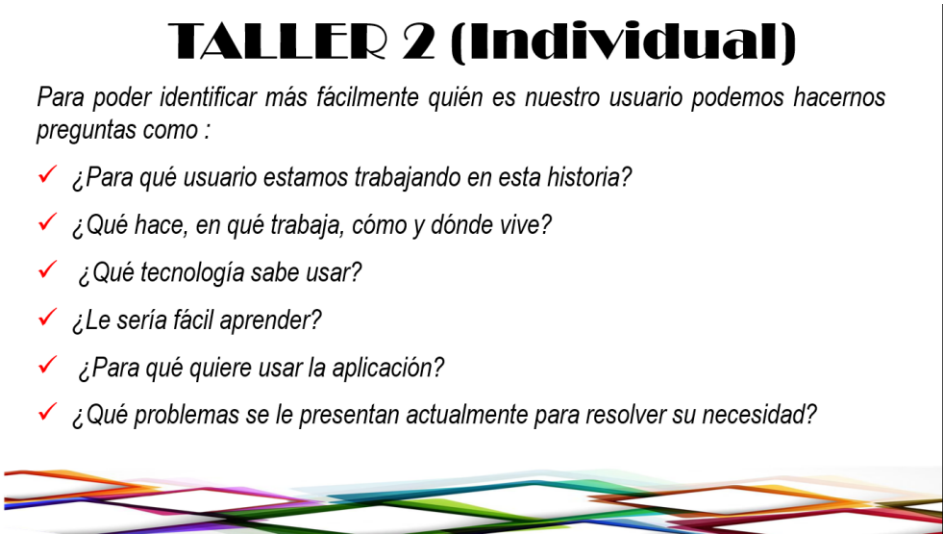


Figura 52. Historia de Usuario

PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

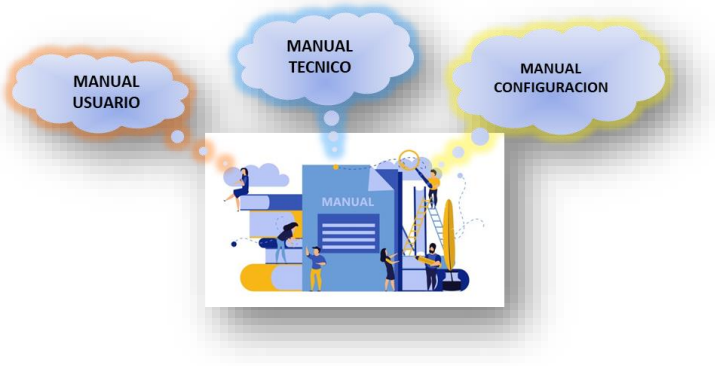
- **Denominación del Programa de Formación:** ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN
- **Código del Programa de Formación:** 2143090
- **Nombre del Proyecto:** DISEÑO Y CONSTRUCCIÓN DE SOFTWARE A LA MEDIDA PARA EL SECTOR EMPRESARIAL
- **Fase del Proyecto:** IMPLANTACION
- **Actividad de Proyecto:** (AP10) DESARROLLAR LAS TAREAS DE CONFIGURACIÓN Y PUESTA EN MARCHA DEL SISTEMA DE INFORMACIÓN.
- **Competencia:** CONSTRUIR EL SISTEMA QUE CUMPLA CON LOS REQUISITOS DE LA SOLUCIÓN INFORMÁTICA.
- **Resultados de Aprendizaje Alcanzar:**
- 22050100702. Elaborar el manual técnico de la aplicación, de acuerdo con la complejidad del aplicativo y según normas y procedimientos establecidos por la empresa.
- **Duración de la Guía:** 20 horas presenciales y 10 autónomas

2. PRESENTACIÓN

Bienvenidos apreciados Aprendices de la Formación Análisis y Desarrollo de Sistemas de Información, esta guía está orientada al manejo conceptual y práctico, necesarias para el desarrollo documental de los procesos técnicos del sistema de información a implementar, donde el aprendiz desarrolle las habilidades para evidenciar a partir del manual técnico los procesos de un sistema de información propuesto, permitiendo difundir, consultar y procesar información de forma interactiva y socializarla a terceros, detectando los errores en el sistema de información y teniendo en cuenta los requisitos documentados del cliente; de igual forma poder elaborar los manuales técnicos del sistema de información, entre ellos el manual de instalación configuración y el manual de usuario, en consonancia de la aplicación.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

3.1 Reflexion:



Manual de los sueños guía para soñar en tiempos difíciles.

Contextualización:

3.2.1 Socializar conceptos básico e importancia de MANUAL TECNICO.pptx suministrado en el material de apoyo.

3.2.2 Socializar ventajas y tipos de MANUALES TECNICOS.pptx

3.2.3 Socializar la importancia y desarrollo de MANUAL TECNICO Y DE USUARIO.pptx

3.2.4 Socializar la importancia y desarrollo de MANUAL DE CONFIGURACION.pptx

3.3 Apropiación:

3.3.1 Atender la formación e indicaciones por parte del instructor líder.

3.3.2 Atender la formación e indicaciones por parte de los instructores de implantación.

3.3.3 Atender la formación e indicaciones por parte de los administrativos SENA.

3.3.4 Diligenciar los formatos de asistencia en cada una de las sesiones de implantación..

3.3.5 Subir el “Taller No. 1 en formato PDF Manual Tecnico con normas APA”

3.3.6 Subir la “Exposición No.1 Guardando la sincronía de colores institucionales SENA ”

3.4 Socialización:

3.4.1 Se revisarán todos los trabajos del taller Nº1 en el ambiente para resolver dudas y dar correcciones a los mismos.

3.4.2 Exponer los temas dados a cada grupo de trabajo para sustentar el conocimiento adquirido.

4. ACTIVIDADES DE EVALUACIÓN

Tome como referencia la técnica e instrumentos de evaluación citados en la guía de Desarrollo Curricular

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento : • **Taller N° 1** • **Exposición de temas** **Evidencias de Producto:**		• Lista de chequeo • Talleres • Test

5. GLOSARIO DE TÉRMINOS

ESTILOS DE APRENDIZAJE: son una mezcla de factores cognitivos, afectivos y fisiológicos característicos que sirven como indicadores relativamente estables de cómo el alumno percibe, interactúa y responde al entorno de **aprendizaje**.

MANUAL TECNICO: Este documento contiene toda la información sobre los recursos utilizados por el proyecto, explicado todo el trabajo que se ha realizado al desarrollar un sistema llevan una descripción muy bien detallada sobre las características físicas y técnicas de cada elemento.

MANUAL DE USUARIO: Es un documento técnico de un determinado sistema que intenta dar asistencia a sus usuarios. En general, un manual de usuario debería poder ser entendido por cualquier usuario principiante, como así también serle útil a usuarios avanzados. Un manual de usuario va dirigido a un público más general, el cual no necesariamente debe tener conocimientos específicos en el área de interés.

MANUAL DE CONFIGURACION: Una especificación o producto que se ha revisado formalmente y sobre los que se ha llegado a un acuerdo, y que de ahí en adelante sirve como base para un desarrollo posterior y que puede cambiarse solamente a través de procedimientos formales de control de cambios.

NORMAS APA: Las normas APA tienen su origen en el año 1929, cuando un grupo de psicólogos, antropólogos y administradores de negocios acordaron establecer un conjunto de estándares o reglas que ayudan a la hora de codificar varios componentes de la escritura científica con el fin de facilitar la comprensión de la lectura.

6. REFERENTES BILBIOGRÁFICOS

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	**Angelica Maria Guerrero Ortiz**	**Instructor**		**6/09/2021**

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

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					

9. Taller Nº1

Este taller tiene como objetivo medir los conocimientos adquiridos con la importancia y desarrollo de Manual Tecnico, para esto se ha dispuesto el material concepto básicos para el desarrollo de esta actividad

Trabajo en clase

1) ¿Qué sabes de manuales?

2) Debate importancia de manuales en desarrollo de software

3) Indique la importancia, y los tipo de manuales técnicos

4) ¿Cuál es la estructura de un manual tecnico

5)¿Cuál es manual técnico mas apto para mi software?

6) ¿En qué consiste el manual de técnico ?

7) Que tipo de manuales técnicos existen

9) Explique las características de un manual tecnico

file:///C:/Users/usuario/Downloads/manualtecnicoymanualdeusuario-121122192452-phpapp02.pdf

https://aleph.org.mx/que-es-un-manual-de-configuracion-de-software#:~:text=Una%20especificaci%C3%B3n%20o%20producto%20que,formales%20de%20control%20de%20cambios.