TRABAJO DE CAMPO 2022

Contenido

[Apuntes de TRABAJO DE CAMPO por Zoel Villar 2](#_Toc100560826)

[Clase 1 2](#_Toc100560827)

[Clase 2 3](#_Toc100560828)

Programas del profe: <https://gitlab.com/UAI-TCTD>

Libro de Pressman

Patrones c#: https://www.dofactory.com/net/design-patterns.

# ACTIVIDADES TRABAJO DE CAMPO

Entrega final: 4/7

* Primer parcial: 16/5
* Primera entrega: 9/5
* Segundo parcial:
* Segunda entrega:

Patrones que implementar:

* ~~Login Logout~~
* Verificar contraseña
* Perfiles
* Encriptación y dígitos verificadores
* Backup
* bitácora
* Multiidioma

# Apuntes de TRABAJO DE CAMPO por Zoel Villar

## Clase 1

Cosas que debe tener el trabajo:

* Login – Logout
* Administración de perfiles
* backUp
* Encriptación
* Dígitos verificadores
* Multi-idioma
* Gestor de bitácora

**Sistemas de información:**

Otorga información, datos procesados (Es diferente al sistema de procesamiento de datos)

* Las BLL deben tener métodos que procesan datos.
* El sistema tiene que pensar por el usuario
* Es claro lo que calcula el sistema, ya que esto no está guardado en el sistema.

**E. de caso de uso:**

Es un documento técnico que contiene al caso de uso, anexos, y mucho más.

**Versiones:**

* Se documenta cada versión
* Las versiones son siempre ejecutables, sino NO es una versión.
* Se utilizan números con significado: V1.1.1 (V1: Modifiqué, 1. Modifiqué x, 1. Corregí ortografía)

**Ingeniería requerimientos:**

Que requerimientos funcionales queremos recolectar.

Se dedica mucho tiempo para el análisis del problema, tiene que estar todo bien definido desde el principio.

Debe tener una evolución, ir creciendo en complejidad, tiene que ser realista en los pasos de los requerimientos.

Primero se puede hacer un prototipo, para poder estar de acuerdo en todos los puntos.

Cambiar algo cuando el sistema ya se encuentra en etapas finales, es sumamente costoso, por lo que hay que tener bien en claro todo desde el principio.

**Requerimiento:**

* Situación o capacidad que se necesita para alcanzar un objetivo
* Pueden ser: Necesidades, deseos o expectativas.
  + Necesidades: Características necesarias para resolver el problema
  + Deseos: Capacidad del sistema que no son imprescindibles, pero si están mejor.
  + Expectativas: Características del sistema esperadas, capaz no explicitas, que se pueden perder fácilmente, como aspectos de la interfaz o de funcionalidad.
* Se deben tener bien en claro estas diferencias, para no confundir como necesario algo que era una boludez.

Tipos de requerimientos:

* Requerimientos de la empresa: Propósitos de la empresa (Quiere ser más eficiente), que tenga ciertos protocolos.
* Requerimientos no funcionales: Que el sistema sea “Rápido”, que sea “Fiable”, Que no se detenga, etc.
* Requerimientos funcionales: Que es lo que el sistema hace. Que procesos son los que genera la información que el usuario necesita.

Entrevista:

* Pensamiento y planificación: Tener objetivos, tener preguntas, tener información de la entrevista, términos y entre otros.
* Conducción de la entrevista: Tomar notas sin imponer ideas o prejuicios, Opinar para abrir puertas. no utilizar adjetivos, lenguaje profesional, ser más objetivo.
* Consolidar y representar la información obtenida.
* Validar la información obtenida

Derivación de escenarios:

Se utiliza el LEL como técnica, interpretar escenarios (Palabras) para una mejor interpretación. (Sujeto, objeto, acción)

**Procesos de RE**

* Elisitación de requerimientos
* Especificación de requerimientos
* Validación de requerimientos.

## Clase 2

**Software:**

El software es el producto que crean los programadores.

Es necesario mantenerlo luego de haberlo largado.

* Obsolescencia: Si los requerimientos cambian, lo que el software hace queda obsoleto, por lo que hay que mantenerlo (Etapa de mantenimiento)
* Etapa de mantenimiento: Intenta extender el tiempo de vida de del software.
* Ser productivo con un software debemos compararnos con lo que ganaríamos en una empresa.

**¿Quién lo hace?:**

* Ingenieros en software
* Arquitectos de software
* Diseñadores
* Analistas
* Programadores

**Importancia:**

* Toda empresa de hoy en día utiliza un mínimo de tecnología para sobrevivir, es su producto organizacional

**Como obtenerlo:**

Está definido por las metodologías

* Tradicionales, Iterativos y evolutivos, agiles
* Se utiliza una diferente en diferentes procesos del trabajo

**Se obtiene:**

* Para un programador:
  + Programas y datos
  + Contribuye al sistema
* Para el usuario:
  + La información
  + El valor agregado a los procesos y negocios de la organización.

**Software como sistema**

El programador no analiza el sistema, el analista es el que lo hace, y le pide al programador las cosas específicas que necesita en el sistema.

**Ingeniería de software:**

Se aplica para poder realizar bien el sistema, son décadas de experiencia, de las cuales se concluye como hacer las cosas bien.

Es un enfoque sistemático, disciplinado y cuantificado al desarrollo, operación y mantenimiento de software

BUENO + BARATO: LENTO

BUENO + RAPIDO: CARO

RAPIDO + BARATO: MALO

El costo de mantenimiento siempre va creciendo.

El costo de mantenimiento del producto no puede sobrepasar el costo de crear el producto desde cero.

Software heredado:

Comúnmente cuando lleguemos a una empresa, no vamos a empezar desde cero, vamos a tener 200 programas ya creados a los que nos vamos a tener que adaptar.

WebApps:

Las aplicaciones webs, necesita uso intensivo de las redes de concurrencia a carga impredecible (pasa de 1000 ventas a 100000), rendimiento (tarda 5 segs en cargar, me voy), disponibilidad, es orientado a datos (Navegadores trabajan con esto), naturaleza sensible, evolución continua, inmediatez, seguridad, estética.

## Clase 3

Encriptación y dígitos verificadores

* Se hace antes de entrar al sistema, si valida abre el sistema.
* Dato extra para verificar la veracidad de los datos. Dato de redundancia
* Se obtiene con un cálculo entre los datos guardados.
* Verificador horizontal para verificar los datos de un row
* Verificador vertical para asegurar que no ingresan o borran un row de datos

Software como proceso:

No es solo escribir código.

¿Qué es un proceso de software?

Texto

Descripción generada automáticamente

* Una herramienta para obtener ese producto con unos estándares de calidad

Tareas principales:

* Comunicación
* Planeamiento
* Moderado
* Construcción
* Despliegue

Tareas secundarias

Flujos de proceso:

* Flujo de proceso lineal

Diagrama

Descripción generada automáticamente

* Flujo de proceso iterativo

Diagrama

Descripción generada automáticamente

* Flujo de proceso evolutivo

Diagrama

Descripción generada automáticamente

* Flujo de proceso paralelo

Diagrama

Descripción generada automáticamente

Entregable: Producto con la idea de.

Patrones de proceso:

Describe un problema de procesos que se da concurrentemente, es una solución estandarizada de personas que ya los estudió

* Nombre
* Fuerzas
* Tipo
* Contexto inicial
* Problema
* Solución
* Contexto resultante
* Patrones relacionados
* Usos y ejemplos conocidos

## Clase 4

Complementa clase 1

**Ingeniería de requerimientos:**

* Es imposible que alguien sepa hacer todo, es necesario contratar gente que sepa del tema, si tengo que hacer un programa de contabilidad, y no se contabilidad, llamo a alguien que sepa del tema, sino no puedo hacer ni los requerimientos.

**Requerimiento:**

Condición o etapa que necesita el usuario para resolver un problema o alcanzar un objetivo

* Funcionales: Que necesita la empresa para cumplir el requisito empresarial.
  + No todo requerimiento es un caso de uso.
* No funcionales: Restricciones al sistema, requerimientos técnicos.
* Empresariales: Razón por la que creo el sistema, lo que la empresa necesita. restricciones del ambiente.

Diagrama

Descripción generada automáticamente

Tipos de requerimientos:

* Necesidades
* Deseos
* Expectativas

Donde están los problemas:

* Falta de input del usuario
* Requerimientos incompletos
* Cambio de los requerimientos
* Expectativas poco realistas
* Falta de claridad
* Practicas no efectivas

A menudo los stakeholders no conocen lo que desean obtener

Se expresan con sus propios términos y con un conocimiento implícito

Distintos stakeholders tienen distintos requerimientos y los expresan de distinta forma

Influencia de los factores políticos

Es inevitable que el ambiente de negocios cambie durante el ciclo de desarrollo

Lo complicado y caro es Mantener el sistema.

**Etapas:**

Diagrama

Descripción generada automáticamente

**Entrevistas:**

**Objetivos de las entrevistas:**

1. Registrar información fácilmente utilizable en los procesos de la RE

2. Descubrir información proveniente del entrevistado precisa y eficientemente

3. Asegurar al entrevistado que su comprensión del tópico ha sido explorado, atendido y valorado

4. La entrevista se planifica, que preguntas hacer, cuanto tiempo va a llevar

Procesos de entrevistas:

**Introducido en 1930 para generar ideas**

Todos los miembros del grupo contribuyen a una lista de prob

resolver o soluciones a un problema

Ayuda a discutir muchas ideas en un corto período de tiempo

Para ser efectivo requiere seguir ciertas reglas:

- establecer un tiempo para la sesión

- no se pueden criticar ideas que se propongan

- asegurar que todos hablen libremente

- asegurar que todos contribuyen

- dejar hablar a los participantes, no al conductor

Ventajas:

- clave para resolver la falta de consenso entre usuarios

- es útil combinarlo con la toma de decisiones

- ayuda a entender el dominio del problema

- encara la dificultad del usuario para transmitir

- ayuda a entender: al usuario y al analista

**BrainStorming:**

Introducido en 1930 para generar ideas

Todos los miembros del grupo contribuyen a una lista de problemas a resolver o soluciones a un problema

Ayuda a discutir muchas ideas en un corto período de tiempo

Para ser efectivo requiere seguir ciertas reglas:

- establecer un tiempo para la sesión

- no se pueden criticar ideas que se propongan

- asegurar que todos hablen libremente

- asegurar que todos contribuyen

- dejar hablar a los participantes, no al conductor

Ventajas:

- clave para resolver la falta de consenso entre usuarios

- es útil combinarlo con la toma de decisiones

- ayuda a entender el dominio del problema

- encara la dificultad del usuario para transmitir

- ayuda a entender: al usuario y al analista

Backup:

Poder hacer una copia de la base de datos, del mdvs y del log

* Voy a tener q bajar el servicio de SQL para hacer la copia

Bitácora:

Registro de que actividades de acciones se llevaron adelante en el sistema

* Se guardan los datos asociados. Un string con los datos del cliente, etc.
* Usuario, rango de fecha, tipo de acción.
* Usuario auditor

MultiIdioma:

Las interfaces se pueden ver en distintos idiomas

* Idioma base, cargar idiomas diferentes,
* En la tabla SQL seria:

|  |  |  |  |
| --- | --- | --- | --- |
| 01 | Botón | Button | Buttoni |

* Cuando inicio sesión, elijo el idioma, voy a la tabla SQL para saber que leyenda se debe mostrar.

**Patrones de diseño:**

(dofactory. C# design pattern)

Patron singleton:

* Bll, clases, métodos de clases
* La clase Tiene que tener un constructor protected

## Clase 5

Borrado lógico: Borrar un objeto de la base de datos no es algo físico, es cambiar una variable para no perder los datos.

Modificación lógica: SI se modifica un precio, se tiene que seguir teniendo un listado del precio histórico de ese producto, lo mismo con la dirección de una persona, si se modifica su dirección, y me preguntan a donde envíe un producto, le tengo que devolver la dirección anterior, no la actual…