

1er Parcial

13/Feb/2024 - Presentación

Maestro: Ary Shared Rosas

Correo: arosasc@ipn.mx 

CIDETEC - Subdirección de incubación

hola

Evaluación

- Examen: 40%
 - Teórico/Practico
 - Equipo
- Proyecto: 60% (Sistema de punto de venta en librería)
 - 3 meses
 - Doc de visión
 - Especificación de requerimientos
 - Código
 - Protocolo de pruebas

Proyecto

-> Tecnología

- Web
- Escritorio
 - > Metodología
- La que queramos
 - > Pasos
- Especificación
- Diseño e implementación
- Validación
- Evaluación

Referencias

- Ian Sommerville
- [Github.com/AryShared](https://github.com/AryShared)

Proyecto punto de ventas - Características

- Interfaz
- Escaner
- Terminal muestre Donde localizarlo y sus existencias
- Portal web (disponible para no asistir)
 - Compra
 - Apartado sin cobro
- Modelo de reportes para monitorear las ventas

Nota: Martes entregar lista de equipos

14/Feb/2024 - C1: Introducción

Sistema:

- Conjunto de cosas que resuelven un problema
- Automatizar
- Forma de administrar
- Regla
- Simplificar cosas complejas



¿Qué es la ingeniería de software?

- Estándares, metodologías y procesos usadas para resolver problemas o necesidades

¿Qué es un buen software?

Accesible, intuitivo, qué sirva, mantenimiento, escalable, sostenible

¿Métodos?

- Cascada
- Espiral
- Scrum
- Kanban

20/Feb/2024

Atributos de software

1. Mantenibilidad
 - Corregir errores

- Actualizar
2. Confianza/Seguridad
 - Seg. informacion
 - Restringir acceso
 - Tolerancia a fallas
 - No falle
 3. Eficiencia
 4. Aceptabilidad
 - Usable
 - Intuitivo

Proceso de software

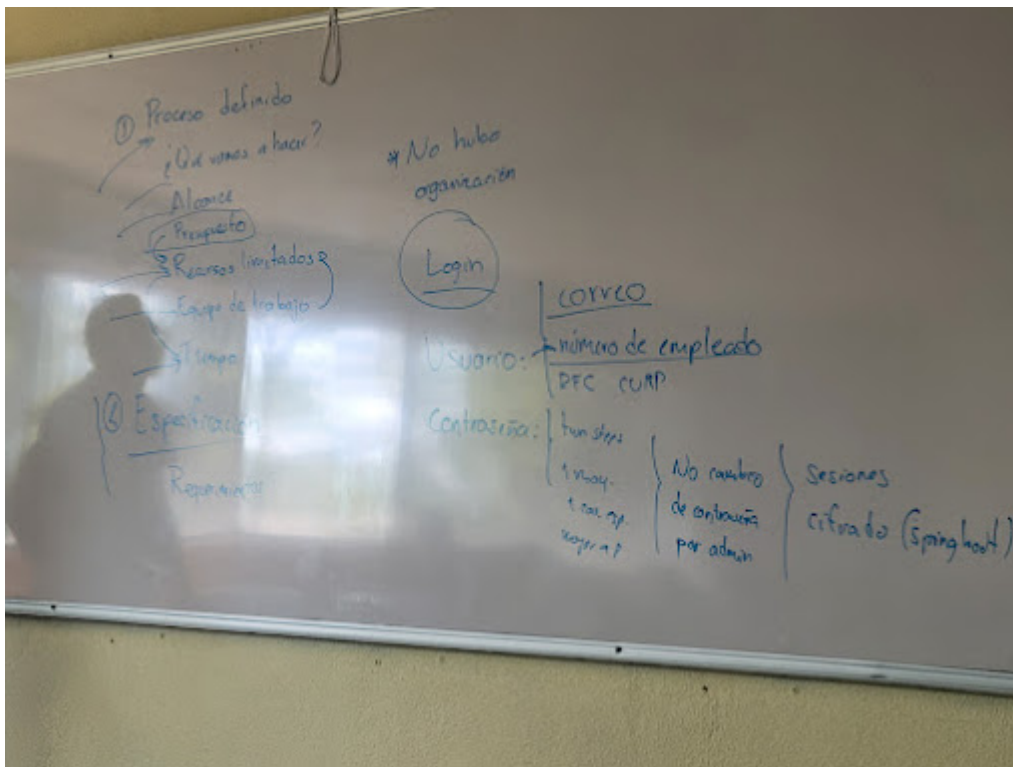
1. Especificación -> ¿Qué quiere el cliente?
2. Diseño e implementación -> Plan/Ejecución
3. Validación ->
4. Evolución

Heterogeneidad -> Elementos diferentes

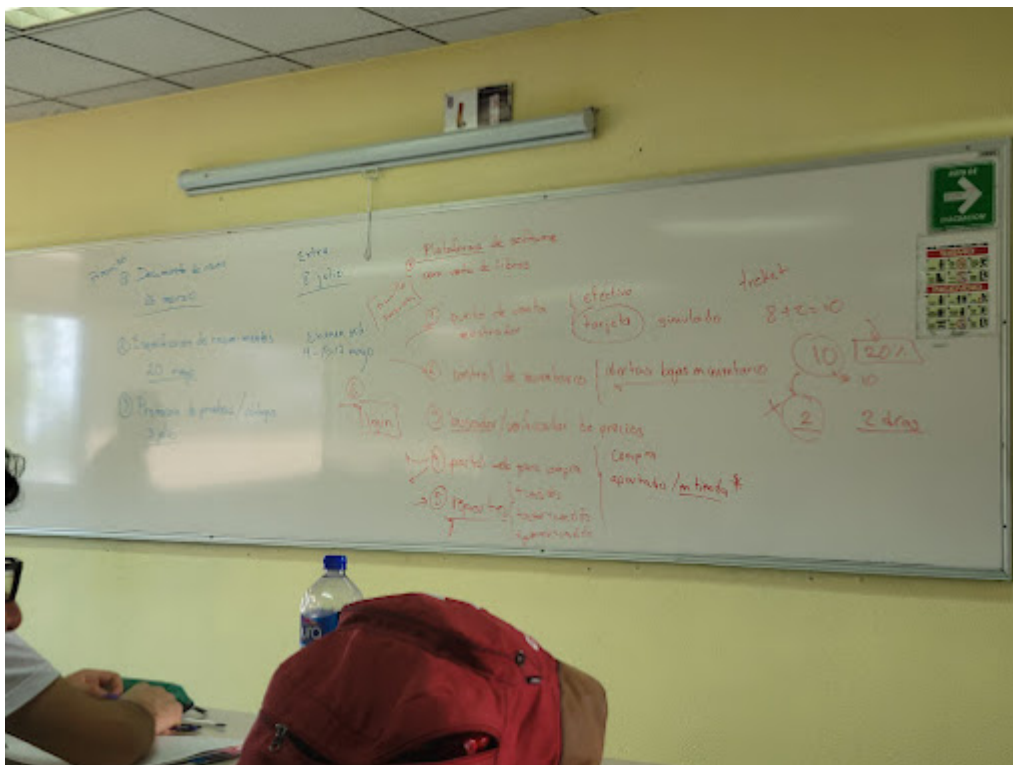
Tipos de software

1. Aplicaciones independientes
2. Aplicaciones transaccionales
3. Entretenimiento
4. Educativo/ingeniería/simulación
5. Sistemas de control embebido
6. Sistemas de adquisición de datos
7. Procesamiento por lotes

21/Feb/2024



23/Feb/2024



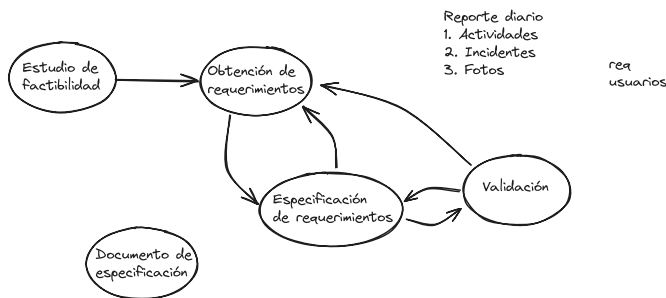
25/Feb/2024

Proceso de software

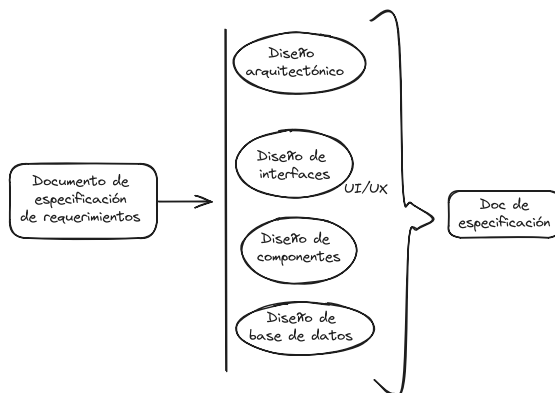
- Serie de pasos para un resultado (software)
- Conjunto de actividades, acciones y tareas que solucionan un problema
- **Especificación**
- **Diseño e implementación**
- **Validación**
- **Evolución**

Especificación

- Requerimientos del sistema: Necesidad específica, solicitado por el cliente, funcionalidad, normas/restricciones.
 - Requerimiento usuario: Lo que se pide
 - Requerimientos sistema: Lo que se hace
- Restricciones
- Planeación
- Documentación de necesidades
- Preguntar al cliente
- Negociación



Diseño



28/Feb/2024

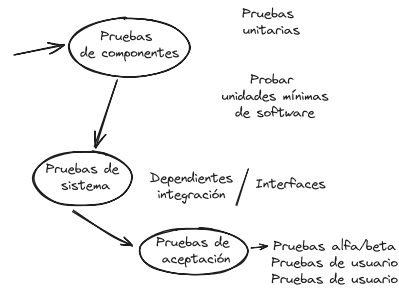
3. Validación / Verificación

¿Construimos el producto correcto?

Es lo que quiere el cliente

¿Construimos correctamente el producto?

Sin errores



4. Evolución

Actualizaciones / Nuevas versiones

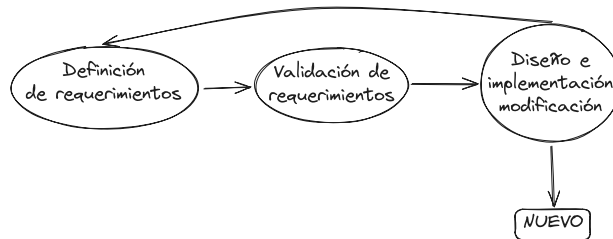
Corrección de errores

Escalamiento

Ampliar la operación

Adaptación

Gestión del cambio



¿Cómo gestionar el cambio?

- Evitar ----> Suceda en momentos importantes. | Anticipación ----> Prototipos
- Tolerancia ----> Metodología Agile

Lineal



Iterativo



Evolutivo



Metodologías

- Cascada
- Desarrollo evolutivo
- RUP
- SCRUM
- KANBAN
- Modelo espiral
- Programación extrema

01/03/2024

CASE - Computer Aided Software Engineering

1. Diagramación

- BDD
- UML
- Star/DIA

2. Generación de reportes (control de versiones)

- git
- svn
- github
- IDE

3. Repositorios centralizados

4. Generadores de documentación

5. Generadores de código (Automatizado)

6. Depuración de código

7. Generación de interfaces de usuario UI/UX

13/03/2024

Análisis de requerimientos

- 1) El cliente sabe lo que quiere? No
- Orientar
 - Descubrir / Qué quiere
 - Qué necesita?

Obtención de requerimientos
¿Qué es un requerimiento?
Descripción de una función del
sistema o una restricción en su operación

Nivel de detalle

Usuario Coloquial

Sistema Específico

- 1) Tipos de actividades
2) Formatos + tipo de datos
3) Origen de los datos
4) Edición, Eliminación, almacenamiento

Requerimientos
- Funcionales
- No Funcionales

Requerimientos no funcionales



Atributos de calidad

- 1) Ambigüedad => Número de modificadores ambiguos
- 2) Volatilidad => Número de cambios
- 3) Compatible => Número de secciones
- 4) Trazable => Número de cambios no detectados
- 5) Usable => Tiempo de capacitación
- 6) Rápido =>

Obtención de requerimientos

- 1) Identificar participantes

2) Considerar diferentes puntos de vista

3) Colaboración

4) Comenzar con preguntas
- Cliente (IPN)

- Usuarios potenciales (Alumnos)

- Ingenieros de software (Del cliente)

- Administradores

- Representantes

- 1) Investigar { Estado del arte
- 2) Preguntar {Entrevistas (Abiertas o cerradas)
- 3) Observar { Etnografías
- { Sistemas existentes

