

Programación Extrema

Ingeniería del Software I

Lic. Laura Gómez Solís

Programación Extrema

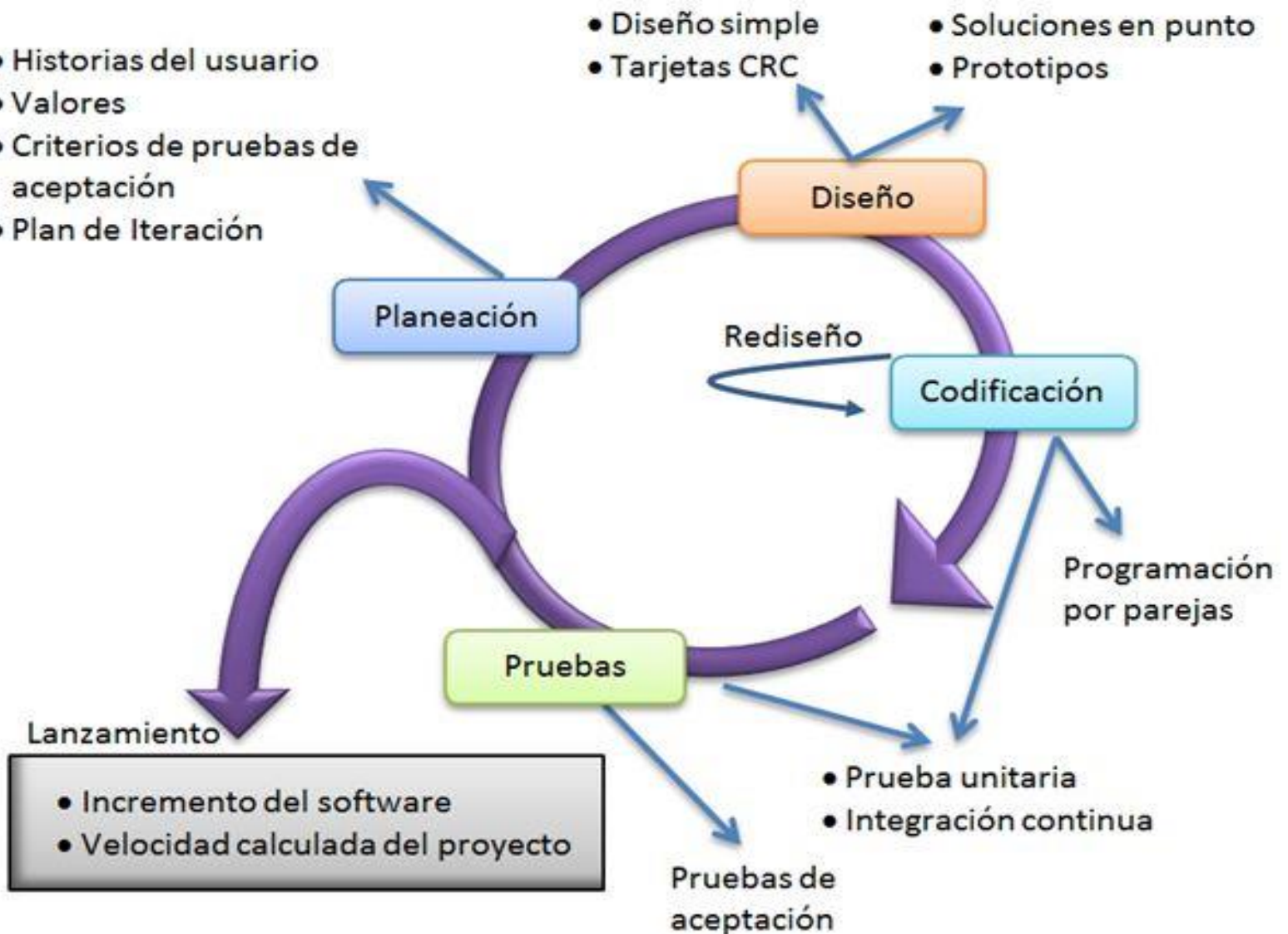
Usa un enfoque orientado a objetos como paradigma preferido de desarrollo, y engloba un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades:

- Planeación
- Diseño
- Codificación
- Pruebas

- Historias del usuario
- Valores
- Criterios de pruebas de aceptación
- Plan de Iteración

- Diseño simple
- Tarjetas CRC

- Soluciones en punto
- Prototipos



Planeación

- Comienza escuchando una actividad para recabar los requerimientos que permite a los técnicos del equipo entender el contexto del negocio para el software y que funcionalidades se requieren.
- Escuchar lleva a la creación de Historias, que son tomadas por los desarrolladores para modelar los requisitos
- Los clientes y desarrolladores trabajan juntos para decidir como agrupar las historias en la siguiente entrega (incremento de software)
- Una vez que se llega a un acuerdo el equipo XP ordena las historias que se desarrollarán en una de 3 formas:

1) Todas las historias se implementarán de inmediato (en pocas semanas),



3) las historias más riesgosas formarán parte de la programación de actividades y se implementarán primero.



2) Las historias con más valor entrarán a la programación de actividades y se implementarán en primer lugar o

Diseño

- sigue rigurosamente el principio mantenelo sencillo.
- El diseño guía la implementación de una historia conforme se escribe, si se encuentra un problema de diseño XP recomienda la creación de un prototipo de esa porción del diseño. Se implementa y se evalúa el prototipo; el objetivo es disminuir el riesgo cuando comience la implementación verdadera.
- En el rediseño el objetivo es controlar las modificaciones.

Tarjetas CRC

- Las tarjetas CRC (Clase – responsabilidad – colaborador) identifican clase relevantes para el incremento actual de software.
- Son el único producto del trabajo de diseño que se genera como parte del proceso XP

NOMBRE DE LA CLASE	
Responsabilidades	Colaboradores

Codificación

- La programación se realiza por parejas creando el código para una historia, a medida que terminan su trabajo el código desarrollado se integra al de los demás.
- Una vez que se codifica una historia se aplica de inmediato la prueba unitaria.



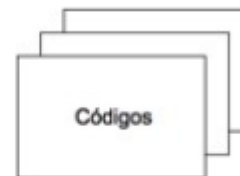
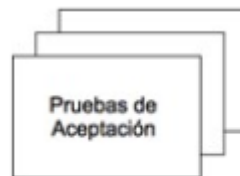
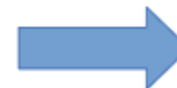
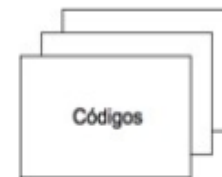
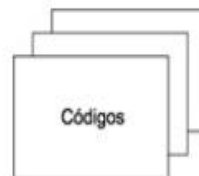
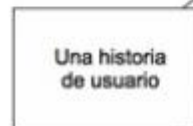
Pruebas

- La creación de pruebas unitarias antes de que comience la codificación es un elemento clave del enfoque de XP, ya que esto asegura la calidad del software



ARTEFACTOS de XP

- Historias de Usuario
- Tareas de Ingeniería
- Pruebas de Aceptación
- Pruebas Unitarias y de Integración
- Plan de Entrega
- Código





El cliente valida las historias y las tareas de acuerdo a las pruebas de aceptación



HISTORIA DE USUARIO

Número: Permite identificar a una historia de usuario.	Usuario: Persona que utilizará la funcionalidad del sistema descrita en la historia de usuario.
Nombre Historia: Describe de manera general a una historia de usuario.	
Prioridad en Negocio: Grado de importancia que el cliente asigna a una historia de usuario.	Riesgo en Desarrollo: Valor de complejidad que una historia de usuario representa al equipo de desarrollo.
Puntos Estimados: Número de semanas que se necesitará para el desarrollo de una historia de usuario.	Iteración Asignada: Número de iteración, en que el cliente desea que se implemente una historia de usuario.
Programador Responsable: Persona encargada de programar cada historia de usuario.	
Descripción: Información detallada de una historia de usuario.	
Observaciones: Campo opcional utilizado para aclarar, si es necesario, el requerimiento descrito de una historia de usuario.	

TAREA DE INGENIERÍA

Número de Tarea: Permite identificar a una tarea de ingeniería.

Número de Historia: Número asignado de la historia correspondiente.

Nombre de Tarea: Describe de manera general a una tarea de ingeniería.

Tipo de Tarea: Tipo al que corresponde la tarea de ingeniería.

Puntos Estimados: Número de días que se necesitará para el desarrollo de una tarea de ingeniería.

Fecha Inicio: Fecha inicial de la creación de la tarea de ingeniería.

Fecha Fin: Final concluida de la tarea de ingeniería.

Programador Responsable: Persona encargada de programar la tarea de ingeniería.

Descripción: Información detallada de la tarea de ingeniería.

PRUEBAS DE ACEPTACIÓN

Código: N° Único, permite identificar la prueba de aceptación.

N° Historia de Usuario: Número único que identifica a la historia de usuario.

Historia de Usuario: Nombre que indica de manera general la descripción de la historia de usuario.

Condiciones de Ejecución: Condiciones previas que deben cumplirse para realizar la prueba de aceptación.

Entrada/Pasos de Ejecución: Pasos que siguen los usuarios para probar la funcionalidad de la historia de usuario.

Resultado Esperado: Respuesta del sistema que el cliente espera, después de haber ejecutado una funcionalidad

Evaluación de la Prueba: Nivel de satisfacción del cliente sobre la respuesta del sistema. Los niveles son: Aprobada y No Aprobada.


Ejemplo – Sistema de biblioteca escolar

Historia de Usuario	
Número: 1	Nombre: Registro de nuevo usuario
Modificación (o extensión) de Historia de Usuario (N° y Nombre): No existe	
Usuario: estudiantes del colegio	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 4
Riesgo en Desarrollo: Bajo	Puntos Reales: 0.5
<p>Descripción:</p> <p>Como estudiante, quiero registrarme en la plataforma de biblioteca digital para acceder al material y realizar préstamos virtuales.</p> <p>Proceso:</p> <ol style="list-style-type: none">1. El usuario accede al sistema y elige “registrarse”2. Ingresa los datos (nombre, DNI, curso, correo institucional)3. El sistema valida los datos y envía correo de confirmación	
Observaciones:	

Tarea de Ingeniería	
Número Tarea: 1	Historia de Usuario (N° y Nombre): # 1 Registro de nuevo usuario
Nombre Tarea: Desarrollo la interfaz	
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Fecha de Inicio: 03/04	Fecha de culminación: 05/04
Programador Responsable: <ul style="list-style-type: none">• Alejandra Matoso• Laura Gómez Solis	
Diseñar un menú que muestre las opciones: Registro de usuario, Cancelar registro.	

Tarea de Ingeniería	
Número Tarea: 2	Historia de Usuario (N° y Nombre): # 1 Registro de nuevo usuario
Nombre Tarea: Validación de correo institucional y DNI único	
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Fecha de Inicio: 06/04	Fecha de culminación: 08/04
Programador Responsable: <ul style="list-style-type: none">• Alejandra Matoso• Laura Gómez Solis	
Verificar que el correo sea del dominio institucional y que el DNI no esté registrado previamente.	

Tarea de Ingeniería	
Número Tarea: 3	Historia de Usuario (N° y Nombre): # 1 Registro de nuevo usuario
Nombre Tarea: Validación de correo institucional y DNI único	
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Fecha de Inicio: 08/04	Fecha de culminación: 08/04
Programador Responsable: <ul style="list-style-type: none"> • Alejandra Matoso • Laura Gómez Solis 	
<p>Enviar un correo institucional confirmando la creación de cuenta con datos de acceso.</p>	



USUARIO	
Registrar Usuario	Usuario
Validar Usuario	

Caso de Prueba de Aceptación	
Código: 1	Historia de Usuario (Nro. y Nombre): # 1 Venta de la tarjeta el Suburbano
Nombre: Prueba módulo registro de nuevo usuario	
Condiciones de Ejecución: El usuario debe acceder al navegador El usuario posee correo institucional El usuario solicita registrarse	
Entrada / Pasos de ejecución: 1.- El usuario solicita registro pide una tarjeta sin saldo 2.- El usuario ingresa los datos solicitados (correo institucional válido y DNI único) 3.- El usuario intenta registro con DNI duplicado 4.- El usuario intenta registro sin correo institucional	
Resultado Esperado: 1.- El sistema muestra formulario de registro. 2.- El sistema valida los datos (correo institucional correcto y DNI único) y registra usuario 3.- El sistema debe rechazar la petición informando el error 4.- El sistema debe rechazar la petición informando el error	
Evaluación de la Prueba: Satisfactorio o no según corresponda	