

Integradora I

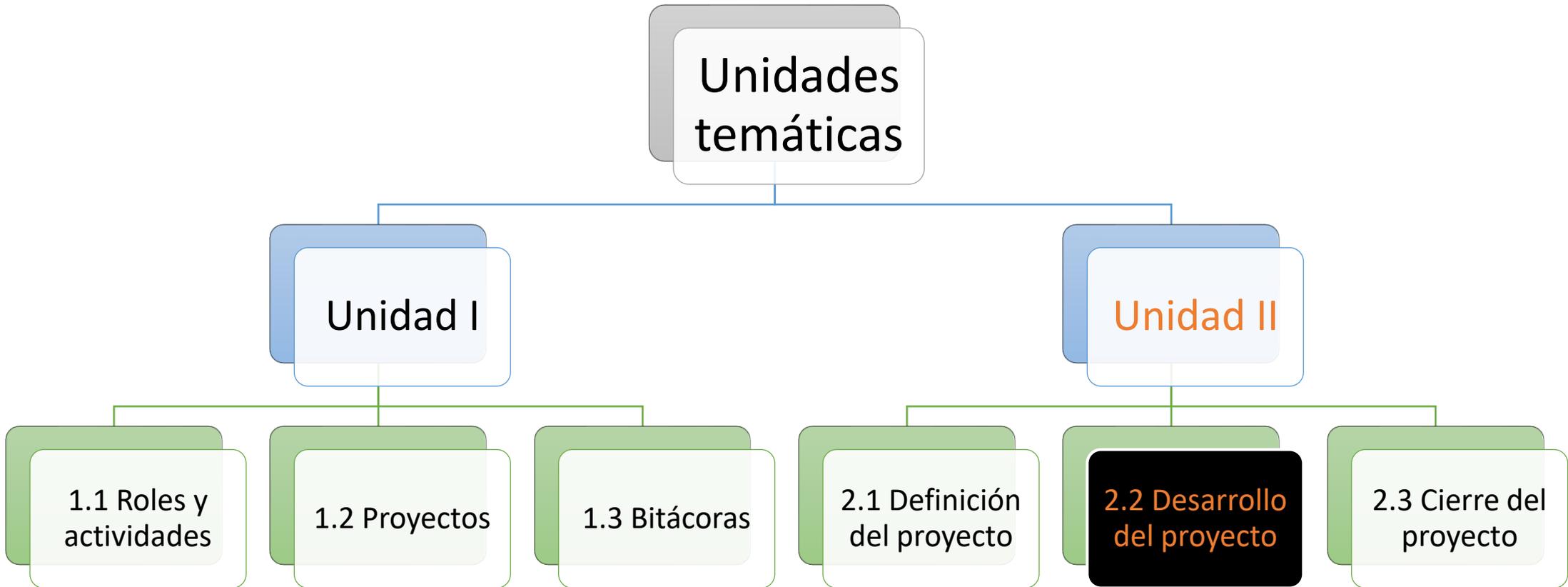
*Unidad II: Fases del proyecto*

*2.2 Desarrollo del proyecto*

*M.G.T.I Brenda Juárez Santiago*

*Periodo: Mayo- Agosto*

# Unidades temáticas



## Índice

UNIDADES TEMÁTICAS _____	2
DESARROLLO DE UN PROYECTO. _____	4
FASES DEL DESARROLLO DEL PROYECTO. _____	4
METODOLOGÍAS UTILIZADAS _____	5
MODELOS DEL PROCESO DEL DESARROLLO DE SOFTWARE. _____	6
MATERIAL DE APOYO. _____	7
REFERENCIAS. _____	8

## Desarrollo de un proyecto.

La definición de etapas de desarrollo de un proyecto consiste en la identificación y organización de todas las actividades y procesos importantes que intervienen en la búsqueda de una meta u objetivo, estas etapas deben ser definidas en función de sus características e importancia que presenten.

Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente. Este proceso es intensamente intelectual, afectado por la creatividad y juicio de las personas involucradas.

### Fases del desarrollo del proyecto.

Cualquier sistema de información va pasando por una serie de fases a lo largo de su vida. Su ciclo de vida comprende una serie de etapas entre las que se encuentran las siguientes:

- Planificación
- Análisis
- Diseño
- Implementación
- Pruebas
- Instalación o despliegue
- Uso y mantenimiento

**Planificación:** Las tareas iniciales que se realizarán esta fase inicial del proyecto incluyen actividades tales como la determinación del ámbito del proyecto, la realización de un estudio de viabilidad, el análisis de los riesgos asociados al proyecto, una estimación del coste del proyecto, su planificación temporal y la asignación de recursos a las distintas etapas del proyecto.

**Análisis:** La etapa de análisis en el ciclo de vida del software corresponde al proceso mediante el cual se intenta descubrir qué es lo que realmente se necesita y se llega a una comprensión adecuada de los requerimientos del sistema (las características que el sistema debe poseer).

**Diseño:** En la fase de diseño se han de estudiar posibles alternativas de implementación para el sistema de información que hemos de construir y se ha de decidir la estructura general que tendrá el sistema (su diseño arquitectónico).

El diseño de un sistema es complejo y el proceso de diseño ha de realizarse de forma iterativa. La solución inicial que proponamos probablemente no resulte la

más adecuada para nuestro sistema de información, por lo que deberemos refinarla.

**Implementación:** Para la fase de implementación hemos de seleccionar las herramientas adecuadas, un entorno de desarrollo que facilite nuestro trabajo y un lenguaje de programación apropiado para el tipo de sistema que vayamos a construir. La elección de estas herramientas dependerá en gran parte de las decisiones de diseño que hayamos tomado hasta el momento y del entorno en el que nuestro sistema deberá funcionar.

**Pruebas:** Errar es humano y la etapa de pruebas tiene como objetivo detectar los errores que se hayan podido cometer en las etapas anteriores del proyecto (y, eventualmente, corregirlos). Lo suyo, además, es hacerlo antes de que el usuario final del sistema los tenga que sufrir.

**Instalación o despliegue:** Una vez concluidas las etapas de desarrollo de un sistema de información (análisis, diseño, implementación y pruebas), llega el instante de que poner el sistema en funcionamiento, su instalación o despliegue.

De cara a su instalación, hemos de planificar el entorno en el que el sistema debe funcionar, tanto hardware como software: equipos necesarios y su configuración física, redes de interconexión entre los equipos y de acceso a sistemas externos, sistemas operativos (actualizados para evitar problemas de seguridad), bibliotecas y componentes suministrados por terceras partes, etcétera.

**Uso y mantenimiento:** La etapa de mantenimiento consume típicamente del 40 al 80 por ciento de los recursos de una empresa de desarrollo de software. De hecho, con un 60% de media, es probablemente la etapa más importante del ciclo de vida del software. (Berzal, 2013)

## **Metodologías utilizadas**

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no llevamos una metodología de por medio, se obtiene clientes insatisfechos con el resultado y desarrolladores aún más.

Con relación a los proyectos que se desarrollan con mayor envergadura, hay si se toma el sentido de basarse en una metodología de desarrollo y se empieza a buscar cuál sería la más apropiada para dicho caso. A fin de cuenta no encontramos muchas veces la más adecuada y se termina por hacer un diseño propio de metodología, por supuesto no está mal siempre y cuando sirva para alcanzar el objetivo.

Algunas Metodologías conocidas:

- La metodología RUP es la más adaptable para proyectos de largo plazo.

- La metodología XP en cambio, se recomienda para proyectos de corto plazo.
- La metodología MSF se adapta a proyectos de cualquier dimensión y de cualquier tecnología.

Se puede decir además que lo más importante antes de elegir la metodología que se debe usar para implementar el software, es determinar el alcance que tendrá y luego de allí ver cuál es la que más se acomoda a la aplicación. (BORONAT, 2018)

### **Modelos del proceso del desarrollo de software.**

No existe consenso sobre cuál es el mejor modelo del proceso software. Distintos equipos de desarrollo pueden utilizar diferentes modelos de proceso software para producir el mismo tipo de sistema software. Sin embargo, algunos modelos son más apropiados para producir ciertos tipos de sistemas, de forma que si no se utiliza un modelo adecuado puede ocurrir que el sistema software resultante sea de menor calidad.

Las características deseables de un proceso de desarrollo software son:

- Claridad: El proceso de desarrollo es claro cuando se entiende con facilidad.
- Visibilidad: Un proceso de desarrollo es visible cuando sus actividades producen resultados claros identificables externamente.
- Facilidad de soporte: Exige disponer de herramientas CASE (Computer-Aided Software Engineering) que den soporte a todas o alguna de las actividades del proceso de desarrollo.
- Fiabilidad: Un proceso de desarrollo es fiable cuando es capaz de detectar posibles errores.
- Facilidad de mantenimiento: Requiere capacidad para incorporar nuevos requisitos o modificar alguno o algunos de los existentes.
- Rapidez: Un proceso software es rápido cuando se puede obtener, a partir de la especificación, una implementación del sistema en un tiempo reducido.

**Modelo en cascada o convencional:** Tomado de otras ingenierías es el primer modelo de desarrollo software propuesto. Ampliamente usado en la industria por su facilidad de gestión y visibilidad.

**Modelo evolutivo:** En este modelo se entrelazan las actividades de especificación, desarrollo y validación. Inicialmente, se desarrolla rápidamente un sistema inicial a partir de una especificación muy abstracta. El sistema se va refinando con la información que van suministrando los clientes y/o usuarios hasta que se obtiene un sistema final que satisfaga todas las necesidades

previstas. El sistema final obtenido puede rediseñarse para producir otro más robusto y más fácil de mantener.

**Modelo transformacional:** Se basa en disponer de una especificación formal del sistema y en transformar, con métodos matemáticos, esta especificación en una implementación. Si las transformaciones que se aplican son correctas es posible asegurar que el sistema construido satisface la especificación, es decir, es posible obtener programas correctos por construcción.

**Modelo en espiral:** El modelo tiene la forma de una espiral en la que cada vuelta representa cada una de las fases en las que se estructura el proceso software y está organizada en cuatro sectores:

1. Definición de objetivos, alternativas y restricciones de cada fase del proyecto.
2. Evaluación de alternativas y análisis de riesgos.
3. Desarrollo y validación. Se elige el modelo de proceso de desarrollo que se considere más adecuado.
4. Planificación de las siguientes fases del proyecto.

(Vigtomann, 2016)

## **Material de apoyo.**

### **Libros.**

Proyectos de Software

<https://www.inf.utfsm.cl/~guerra/publicaciones/Gestion%20de%20Proyectos%20de%20Software.pdf>

Administración de proyectos de Software

[http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas\\_Admon\\_de\\_Proyectos\\_v2\\_2.pdf](http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas_Admon_de_Proyectos_v2_2.pdf)

### **Videos.**

<https://youtu.be/VdZRNuB4p7g> Modulo 4. Desarrollo de un proyecto de TI

<https://youtu.be/BmSOFC0I10M> **Cómo mejorar la gestión de proyectos y emprendimientos con Luis Font**

<https://youtu.be/zysGUTaPMTk> **Gestión de Proyectos**

## Referencias.

<http://elvex.ugr.es/idbis/db/docs/lifecycle.pdf>

[https://www.ecured.cu/Desarrollo\\_de\\_software](https://www.ecured.cu/Desarrollo_de_software)

<http://www.voigtmann.de/es/desarrollo-de-software/>

<http://www.tesoem.edu.mx/alumnos/cuadernillos/2011.009.pdf>

[http://arantxa.ii.uam.es/~proyectos/teoria/C5\\_Proyectos%20de%20desarrollo%20software.pdf](http://arantxa.ii.uam.es/~proyectos/teoria/C5_Proyectos%20de%20desarrollo%20software.pdf)

Berzal, F. (2013). *El ciclo de vida de un sistema*. Obtenido de Diseño de bases de datos:  
<http://elvex.ugr.es/idbis/db/docs/lifecycle.pdf>

BORONAT, A. (25 de Mayo de 2018). *Ecured*. Obtenido de Desarrollo de software:  
[https://www.ecured.cu/Desarrollo\\_de\\_software](https://www.ecured.cu/Desarrollo_de_software)

Vigtomann. (2016). *Vigtomann*. Obtenido de Desarrollo de sw:  
<http://www.voigtmann.de/es/desarrollo-de-software/>