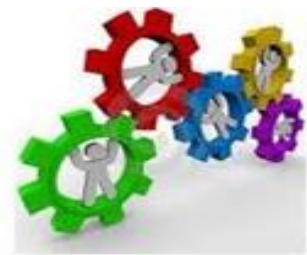


**GUÍA DIDÁCTICA  
UNIDAD CURRICULAR  
PROYECTO SOCIO TECNOLÓGICO  
PROGRAMA NACIONAL EN INFORMÁTICA**



*“Enseñar no es transferir conocimiento, sino crear las posibilidades para su propia producción o construcción”*  
Paulo Freire

**Autor Lcda. Msc Mirayda Canela**

Acarigua, Mayo de 2023

	ÍNDICE	Pg.
<b>INTRODUCCIÓN</b>		<b>4</b>
<b>PRIMER COMPONENTE - Identificación del Curso</b>		<b>6</b>
<b>SEGUNDO COMPONENTE - Objetivos</b>		<b>8</b>
<b>TERCERO COMPONENTE - Contenido</b>		<b>9</b>
<b>PRIMERA PARTE – Elaborando el Modelo del Negocio</b>		<b>10</b>
<b>Objetivos</b>		<b>10</b>
<b>Temas</b> 	El Proyecto Socio Tecnológico Planificación de proyecto Equipo del proyecto Técnicas e instrumentos de recolección de datos. Investigación Acción Participativa Marco Lógico Estudio de Factibilidad MERINDE Modelo del Negocio Lenguaje Unificado de Modelaje (UML).	<b>11</b> <b>12</b> <b>13</b> <b>17</b> <b>19</b> <b>20</b> <b>23</b> <b>25</b> <b>27</b> <b>35</b>
<b>Actividades</b>		<b>38</b>
<b>Evaluación</b>		<b>40</b>
<b>SEGUNDA PARTE – Elaborando el Modelo del Sistema</b>		<b>50</b>
<b>Objetivos</b>		<b>51</b>
<b>Temas</b> 	Requerimientos del Software Diagrama de Caso de Uso del Software Arquitectura del Software Modelo de Base de Datos Modelo de Diseño Mapa de Navegación. Interfaz de Usuario Diagrama de Componente Diagrama de Despliegue	<b>52</b> <b>55</b> <b>57</b> <b>58</b> <b>60</b> <b>63</b> <b>65</b> <b>67</b> <b>68</b>
<b>Actividades</b>		<b>72</b>
<b>Evaluación</b>		<b>73</b>
<b>TERCERA PARTE - Implantando el Software</b>		<b>79</b>

<b>Objetivos</b>	<b>80</b>
<b>Temas</b>	<b>81</b>
	
Pruebas del Software	86
Estrategia de Implementación	90
Factores de Riesgo	91
Documentación del Software	91
<b>Actividades</b>	<b>94</b>
<b>Evaluación</b>	<b>95</b>
<b>CUARTO COMPONENTE - Evaluación del Curso</b>	<b>98</b>
<b>Objetivos</b>	<b>99</b>
<b>Temas</b>	<b>100</b>
	
Planificación Académica y de Evaluación	102
Ponderación de Evaluaciones	103
Seguimiento del Proyecto	103
<b>Actividades</b>	<b>107</b>
<b>Evaluación</b>	<b>108</b>
<b>QUINTO COMPONENTE - BIBLIOGRAFIA</b>	<b>110</b>
<b>GLOSARIO</b>	<b>112</b>
<b>ANEXOS</b>	<b>113</b>
<b>A: FICHA REGISTRO DEL PROYECTO</b>	<b>113</b>

## INTRODUCCIÓN

**La Unidad Curricular Proyecto Socio Tecnológico (UC-PST),** núcleo central de formación, permite al participante vivenciar su rol profesional durante el desarrollo del Proyecto Socio tecnológico (PST). Sobre este escenario, el PST está orientado a ofrecer productos y servicios a la Comunidad, con objeto de dar solución a problemas reales. Exige la vinculación de los saberes que convergen en el trayecto y un trabajo colaborativo entre los estudiantes, profesor asesor y comunidad.

Ahora bien, el docente asesor debe emprende un seguimiento durante los dos semestres del trayecto, para concretar que los equipos de estudiantes implanten en la comunidad y socialicen el producto que se desprende del PST. Por consiguiente, se traza estrategias fungidas en un plan metodológico de desarrollo, con entregas parciales, puntos de control hasta la versión final.

Esta obra titula: GUIA DIDÁCTICA, PARA EL FORTALECIMIENTO DE LA UC-PST, tiene como finalidad la orientación del curso de forma sencilla, y amigable. Está estructurado en cinco (5) componentes. Cada uno de ellos, suministra aspectos teóricos y prácticos ajustados a la dinámica del curso. Los referentes que dan formalidad a la idea se presentan estructurados como se especifica:

**PRIMER COMPONENTE,** en este apartado se indica la naturaleza del curso mediante su nombre, área, nivel, duración modalidad de estudio, línea de investigación, orientaciones pedagógicas, forma de evaluación y de iteración.

**SEGUNDO COMPONENTE,** expone los objetivos generales del curso, los cuales orientan las metas por concretar.

**TERCER COMPONENTE – Contenido:** en este apartado se presenta una serie de temas agrupados en tres (3) partes de forma lógica durante el desarrollo del proyecto:

**LA PRIMERA PARTE - Elaborando el Modelo del Negocio:** en este se ubican los temas: técnicas de levantamiento de información, fases del diagnóstico participativo, marco lógico, definición del problema, estudio de factibilidad, planificación de proyecto, modelo del negocio, diagrama de caso de uso del negocio y diagrama de actividad.

**LA SEGUNDA PARTE – Elaborando el Modelo del Sistema:** en este apartado se señalan los temas referidos a: requerimientos del software, diagrama de caso de uso del software, arquitectura del software, modelo de base, de dato, modelo de diseño, mapa de navegación, interfaz de usuario, diagrama de componente y diagrama de despliegue.

**EN LA TERCERA PARTE –Evaluando la implantación del Software:** es donde se plantean los temas: pruebas del software, estrategia de implantación, factores de riesgo y documentación del software.

**CÓMO CUARTO COMPONENTE – Evaluación del Curso:** apartado donde se expresa la forma de evaluación del curso durante el trayecto.

**PARA FINALIZAR EL QUINTO COMPONENTE BIBLIOGRAFIA:** en donde se sugiere obras literarias digitales y físicas que pueden consultar sobre los temas del curso.

# PRIMER COMPONENTE

## IDENTIFICACIÓN DEL CURSO

<b>NOMBRE</b>	<b>Unidad Curricular Proyecto</b>
<b>NIVEL</b>	TRAYECTO II, TRAYECTO III y TRAYECTO IV
<b>AREA</b>	Proyecto Socio Tecnológico
<b>DURACIÓN</b>	Dos (2) semestres.
<b>MODALIDAD DE ESTUDIO</b>	Multimodal
<b>LÍNEA DE INVESTIGACIÓN</b>	Desarrollo de Software
<b>ORIENTACIONES PEDAGOGICAS</b>	<p><b>El docente puede realizar:</b></p> <ul style="list-style-type: none"><li>• Exposiciones con el contenido.</li><li>• Socializar los Avances.</li><li>• Evaluar competencias.</li><li>• Orientar el desarrollo del PST.</li></ul> <p><b>El estudiante puede realizar</b></p> <ul style="list-style-type: none"><li>• Socializar con los compañeros el contenido.</li><li>• Exponer el contenido.</li><li>• Diagnosticar sus competencias.</li><li>• Orientar el desarrollo del PST.</li></ul> <ul style="list-style-type: none"><li>• Asesorías</li><li>• Avances del Informe.</li><li>• Puntos de Control</li><li>• Pre socialización</li><li>• Socialización</li></ul>
<b>EVALUACIÓN</b>	<ul style="list-style-type: none"><li>• Presencial</li><li>• Correo Electrónico.</li><li>• Grupo virtual de Asesorías.</li></ul>
<b>ITERACIÓN</b>	

## PRIMER COMPONENTE

### DESCRIPCIÓN ICONOS UTILIZADOS

ICONO	SIGNIFICADO
	Indica objetivos
	Temas
	Actividades
	Evaluaciones
	Consejo Práctico.
	Avance
	Actividad en Grupo
	Evaluación
	Asesoría
	Inicio de un tema
	Demostración
	Lectura
	Actividad de Aprendizaje

## SEGUNDO COMPONENTE

### OBJETIVOS DEL CURSO

**Al finalizar el curso,** el estudiante será capaz de:

Reconocer los conceptos básicos en el desarrollo de un software.

Describir la importancia del uso de la metodología en el desarrollo del software.

Diseñar los artefactos UML que se expresan durante el desarrollo del software



Elaborar el Modelo del Negocio, para abordar el diseño del software.

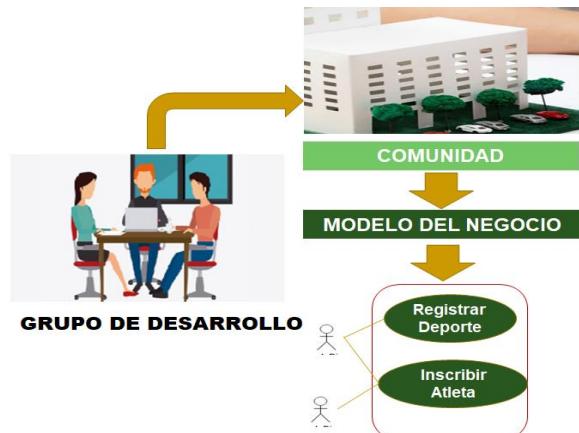
Elaborar el Modelo del Sistema, para abordar la construcción del software

Emprender la implantación del software, para evaluar la calidad del software

## TERCER COMPONENTE

### CONTENIDO

## PRIMERA PARTE ELABORANDO EL MODELO DEL NEGOCIO



Consiste en comprender la realidad del negocio desde sus objetivos, estructura, procesos, recurso humano, documentación, recursos tecnológicos y problemáticas detectadas. Este describe el qué, porqué y cómo del negocio.

En él se presentan artefactos UML, para una comprensión universal de los profesionales del área de la informática y para los usuarios.

## ELABORANDO EL MODELO DEL NEGOCIO

### OBJETIVO(S)



- **Constituir el equipo de trabajo**, para asignar las responsabilidades en el desarrollo del producto.
- **Identificar la comunidad** u organización, para ubicar el objeto de estudio.
- **Realizar el diagnóstico participativo**, para identificar la situación que requiere una solución informática.
- **Determinar la factibilidad** del desarrollo del producto, para abordar el desarrollo de la solución.
- **Planificar las actividades**, para abordar la solución de la problemática planteada de forma sistematizada y con responsabilidades.
- **Presentar el modelo del negocio** sobre los argumentos metodológicos de MERINDE y la notación UML, para conocer el escenario actual de (los) proceso(s) en estudio.

### TEMAS



- 1.1 El Proyecto Socio Tecnológico.
- 1.2 Planificación de Proyecto.
- 1.3 Equipo del proyecto.
- 1.4 Técnicas e Instrumentos de Recolección de Datos.
- 1.5 Investigación Acción Participativa.
- 1.6 Marco Lógico.
- 1.7 Estudio de Factibilidad.
- 1.8 MERINDE.
- 1.9 Modelo del Negocio.
- 1.10 UML.

### ACTIVIDADES

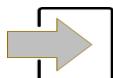


- Asesoría.
- Lectura UML / MERINDE.
- Infografía - Modelo del Negocio.
- Registro en Ficha Equipo de Trabajo

## PRIMERA PARTE

### ELABORANDO EL MODELO DEL NEGOCIO

#### TEMAS



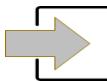
**EL PROYECTO SOCIO TECNOLÓGICO (PST):** el Plan Rector del PNFI 2008 expone: "es el núcleo central del PNFI, garante de la construcción del conocimiento y del desarrollo del pensamiento crítico, reflexivo, abierto, dinámico y productivo...". En este orden de ideas, se identifica como una estrategia de aprendizaje a partir del aprender haciendo. En este contexto, se logra desarrollando un software, dando solución a una problemática real detectada en una comunidad. Exige la **vinculación de los conocimientos que se desprenden de las unidades curriculares Ingeniería del Software, Base de Datos, Programación, Redes, entre otras.** (Ver imagen 1).



Imagen 1: Contexto del PST. Fuente: Pizarrón Didáctico. Canela, M (2020).



Indica según la experiencia del trayecto anterior, **la importancia del proyecto socio tecnológico en tu formación profesional.**



**PLANIFICACIÓN DE PROYECTO:** La planificación de un proyecto, tiene por finalidad crear un **plan de trabajo para acompañar el progreso del equipo**. Harold Kerzhen (2003) expone: “La planificación del proyecto es la ordenación sistemática de las tareas para lograr un objetivo, **expone lo que se necesita hacer y cómo debe llevarse a cabo**”. (p.104). Por lo tanto, es fijar un curso concreto de acciones, estableciendo secuencia de actividades, tiempo, responsables, recursos y resultado esperado.



En el contexto de **UC-PST**, se vinculan acciones para la puesta en marcha de conocimientos metodológicos indicados en la investigación acción participativa, marco lógico, la metodología de desarrollo de software MERINDE y la notación para el diseño de diagramas UML. (Ver **cuadro 1, Plantilla para presentar la planificación del proyecto** según MERINDE).

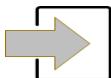
**Cuadro 1: Plantilla – Planificación de Proyecto.**

FASE I-INICIO					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS
FASE II – ELABORACIÓN					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS
FASE III – CONSTRUCCIÓN					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS
FASE IV – TRANSICIÓN					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS

Fuente: Canela, M (2023).



**Ubica la planificación de la unidad curricular**, el esquema que orienta la presentación del informe; esto para orientar las metas según los cortes y duración del semestre.



**EQUIPO DE PROYECTO:** es un conjunto de personas multidisciplinarias, la cual comparten sus conocimientos para alcanzar los objetivos del proyecto.



**En el contexto del PST**, lo constituyen un grupo de estudiantes, el profesor asesor y representantes de la comunidad, **quienes asumirán roles en el proyecto**; es importante tomar en cuenta las habilidades y destrezas de cada integrante para la designación.



**Identifica tu grupo de PST:** pregunta al profesor de la UC-PSTST, la cantidad de integrantes por grupo y selecciona con que compañeros de estudio deseas trabajar. Toma sus datos personales: nombre, apellido, teléfono y correo electrónico. **Organiza un grupo virtual para compartir ideas.**

Ahora bien, Villareal, C (2016), **define Rol** como “la responsabilidad individual de cada participante en un proyecto, cada uno es un pilar fundamental que sostiene otro, y que trabajan en conjunto para lograr una meta”. (p.1). A su vez, según la **Metodología De la Red Nacional de Integración y Desarrollo de Software Libre (MeRinde)** lo define como: “Las responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo.”

## ESPECIFICACIÓN DE LOS ROLES:



**Analista de Producto:** Se encarga de dirigir el proceso de captura de requerimientos, definir los actores y casos de uso y estructurar el modelo de casos de uso, estableciendo la forma en que funcionará el sistema y cuáles son las restricciones del mismo.



**Analista de calidad:** Se encarga de revisar todos los documentos que reflejan el avance del proyecto (diagrama Gantt, reporte de estado, actas de reunión, reporte de pendientes, y otras afines al control y seguimiento del proyecto), y de verificar que los objetivos del marco de desarrollo se cumplan.

**Arquitecto del software:** se encarga de la definición de la arquitectura que guiará el desarrollo, y de la continua refinación de la misma en cada iteración; debe construir cualquier prototipo necesario para probar aspectos riesgosos desde el punto de vista técnico del proyecto; definirá los lineamientos generales del diseño y la implementación.



**Líder del Proyecto:** tiene la función de dirigir y asignar recursos, coordina las interacciones con los clientes y usuarios finales, planifica las iteraciones, planifica y asigna el trabajo, define la organización del proyecto, establece las prácticas que aseguran la integridad y calidad de los artefactos del proyecto, entre otras responsabilidades.



**Desarrollador:** corresponde a la persona que tiene a su cargo la codificación de los componentes en código fuente en algún lenguaje de programación en la que debe elaborar y ejecutar las pruebas unitarias realizadas sobre el código desarrollado.



**Involucrados:** Cualquier persona que se vea afectada por el resultado del proyecto es considerada como un involucrado. Comprende un grupo de personas interesadas en que sus necesidades sean satisfechas por el proyecto.



**Probador:** la función del probador es realizar las pruebas identificadas y definidas previamente, utilizando las instrucciones, métodos y herramientas necesarias para este rol. Debido a la realización de las pruebas debe obtener los resultados de las mismas.



**Mentor:** es aquella persona que está íntimamente ligada con el proceso de desarrollo de software, que conoce todas las prácticas involucradas y entiende el porqué de la misma. Acompaña y apoya a los equipos de trabajo mediante revisiones de los artefactos y haciendo recomendaciones de cómo mejorar los mismos durante todo el ciclo de vida del sistema.



**Asigna el rol a cada integrante de tu grupo.** Todo el equipo trabajando unido en el cumplimiento de los objetivos del proyecto.



Sobre el escenario de PST, **se sugiere que cada estudiante debe vivenciar todos los roles durante PST II, PST III y PST IV**. El estudiante debe dar su mayor contribución al proyecto y por ende sumar experiencias en su formación profesional. El equipo de trabajo debe presentarse según el **cuadro 2**:

**Cuadro 2:** Asignación de roles.

ROL	DESCRIPCIÓN	RESPONSABLES
ANALISTA DE CALIDAD	<< indicar que responsabilidad tiene que cumplir el rol>>	<< colocar nombres estudiantes asignados>>
ANALISTA DE PRODUCTO.		<< colocar nombres estudiantes asignados>>
ARQUITECTO DE SOFTWARE.		<< colocar nombres estudiantes asignados>>
DESARROLLADOR.		<< colocar nombres estudiantes asignados>>
INVOLUCRADOS.		<< colocar nombres estudiantes, comunidad, profesor asesor>>
LÍDER DEL PROYECTO.		<< colocar un solo estudiante>>
PROBADOR		<< colocar nombres estudiantes y representantes comunidad>>
MENTOR		<< colocar nombres profesor asesor/tutor/>>

Fuente: Canela (2023).

**LA ORGANIZACIÓN:** Es un grupo social compuesto por **personas, tareas, procesos y una administración que forman la estructura sistemática de relaciones de interacción**; todo con el objeto en producir bienes y servicios, para satisfacer las necesidades de una comunidad dentro de un entorno, y así poder lograr el propósito distintivo que es su misión. Thompson (2007) define una organización como: “es un sistema cuya estructura está diseñada para que los recursos humanos, financieros, físicos, de información y otros, de forma coordinada, ordenada y regulada por un conjunto de normas, logren determinados fines”. (p.1).

### Organización



### Organigrama Estructural



## Recurso Humano



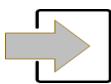
## Objetivos Estratégicos



Se sugiere al grupo del **PST**, ubicar una comunidad u **organización que esté bien constituida**, es decir, exista un organigrama estructural que indique departamentos, objetivos estratégicos, procesos, personal a cargo, entre otros como manual de normas de procedimientos y tecnología (computadores/red local/servicio de internet). En su recorrido para ubicarla, debe apoyarse en las técnicas e instrumentos de recolección de información.



**Identifica tu Comunidad.** Uds. estudiantes como equipo, realicen una lista de organizaciones que puedan visitar y emprender el proyecto.



## TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN

**DATOS:** son herramientas que permite al investigador realizar el reconocimiento de un objeto de estudio, desde la obtención de información que pueda ser resguardada en un medio material físico o digital para su posterior interpretación.

**Técnicas de recolección de datos:** según Arias, F (2006) “**es el procedimiento o forma particular de obtener datos o información. Es un complemento que va de la mano con una investigación**”. (p.67).

Por lo tanto, en este contexto, los estudiantes deben utilizar una o varias técnicas, para obtener toda la información sobre comportamiento de los procesos, documentos involucrados, roles de las personas involucradas en el mismo, recursos tecnológicos presentes, estructura de la organización, entre otros. Es decir, debe hacer el reconocimiento del contexto que abriga del objeto de estudio.

**Instrumento de recolección de datos:** según Arias, F (2006) “**es cualquier recurso, dispositivo o formato (físico o digital), que se utiliza para obtener, registrar o almacenar información relevante a la investigación**”. (p.69).



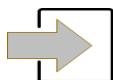
En la **imagen 2**, puedes observar la relación de técnicas e instrumentos de recolección de datos; conceptos fundamentales, los cuales serán aplicados durante toda las fases del PST.



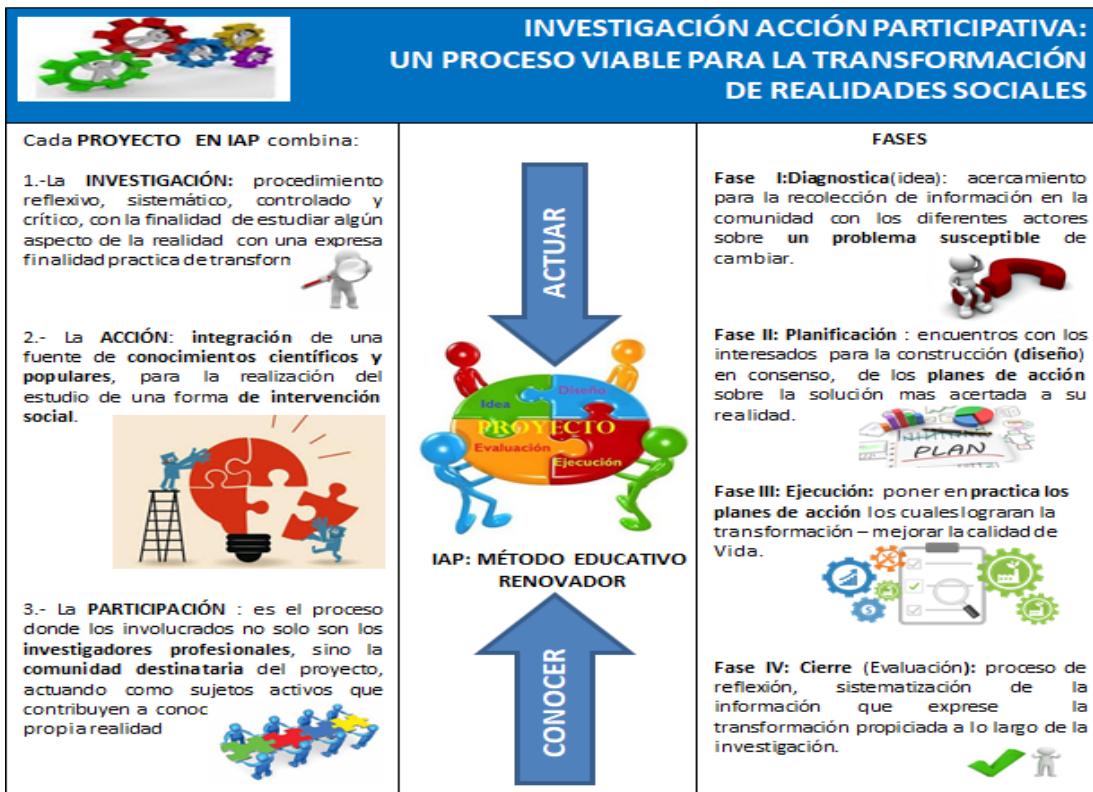
**Describe tu Comunidad.** Uds. como grupo de proyecto, seleccionen las técnicas que van a usar para el levantamiento de información y diseñen los instrumentos. Verifica previamente que información se encuentra enmarcada en el esquema de informe

TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS: CONCEPTOS QUE CONOCER, PARA EMPRENDER UN PROYECTO / INVESTIGACIÓN DE CAMPO.	
TÉCNICAS	INSTRUMENTOS
	<b>OBSERVACIÓN:</b> consiste en visualizar o captar mediante la vista, en forma sistemática, cualquier hecho, fenómeno o situación que se produzca en la naturaleza o en la sociedad, en función de unos objetivos pre establecidos.  <b>ENCUESTA:</b> pretende obtener información que suministra un grupo o muestra de sujetos acerca de si mismos, o en relación con un tema particular, y objeto concreto.  <b>ENTREVISTA:</b> basada en el dialogo o conversación "cara a cara", entre el entrevistador y el entrevistado, acerca de un tema previamente determinado,
	<b>Estructurada</b> Lista de Cotejo, Escala de Estimación.  <b>No Estructurada</b> Diario de Campo, Cámaras fotográficas y de videos
	<b>Oral</b> Grabadora, guía de la encuesta, cámara de video  <b>Escrita</b> Cuestionario
	<b>Estructurada</b> Guía de entrevista, grabador/cámara de video.  <b>No Estructurada</b> Libreta de Notas, grabador/cámara de video.

**Imagen 2:** Técnicas e Instrumentos de Recolección de Datos. Fuente: Pizarrón Didáctico elaborado por Canela, M (2020).



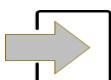
**INVESTIGACIÓN ACCIÓN PARTICIPATIVA (IAP):** Según Salazar (2006), la IAP es “nueva visión de la sociedad, el conocimiento y la ciencia; trabajando y aprendiendo en acciones de tipo colectivo, en una mejor posibilidad de transformación de una situación.” (p.57). Es decir, se articulan alternativas de solución ante una realidad social que exige mejor calidad de vida y se logra a través de la vinculación del conocimientos en la construcción, la creación y el vivir. **En la imagen 3,** se ilustra el contexto teórico de la IAP.



**Imagen 3,** Contexto teórico de la IAP. Fuente: Pizarrón Didáctico, elaborada por Canela, M (2020).

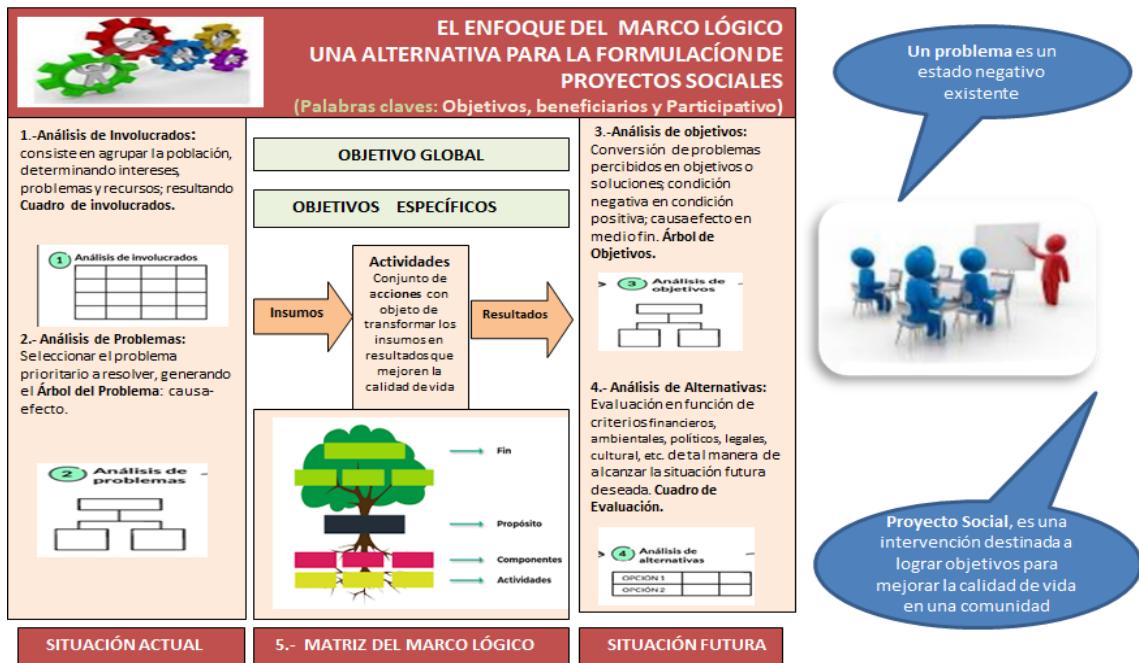
El **PST** subyace en este principio, ya que ofrece un producto para la comunidad; donde el conocimiento es aportado por los involucrados hasta concretar los objetivos y la puesta en marcha del mismo.

Determina con tus compañeros la **importancia de la investigación acción participativa en su proyecto.**



**MARCO LÓGICO:** según Cortés, D (1997) “es una herramienta analítica para la planificación y la gestión de proyectos orientadas por

objetivos" (p.13). Esta metodología incluye representantes de las partes involucradas en el diseño de los artefactos (análisis de involucrados, árbol del problema, árbol de objetivos, análisis de alternativas de solución) que van estructurando la solución ante la problemática detectada.(Ver **Imagen 4**).



**Imagen 4:** Contexto teórico del Enfoque del Marco Lógico. Fuente: Pizarrón Didáctico, elaborada por Canela, M (2020).

 En el contexto del **PST**, del **marco lógico**, se sugiere diseñar los artefactos para el reconocimiento de los involucrados, árbol del problema y árbol de objetivos.

**ARBOL DEL PROBLEMA:** según la UNESCO (2017) señala: "El árbol de problemas, es una técnica que se emplea para identificar una situación negativa (problema central), la cual se intenta solucionar analizando relaciones de tipo causa-efecto" (p.1). En la imagen 8 se observa el contexto teórico que arropa el diseño del árbol del problema

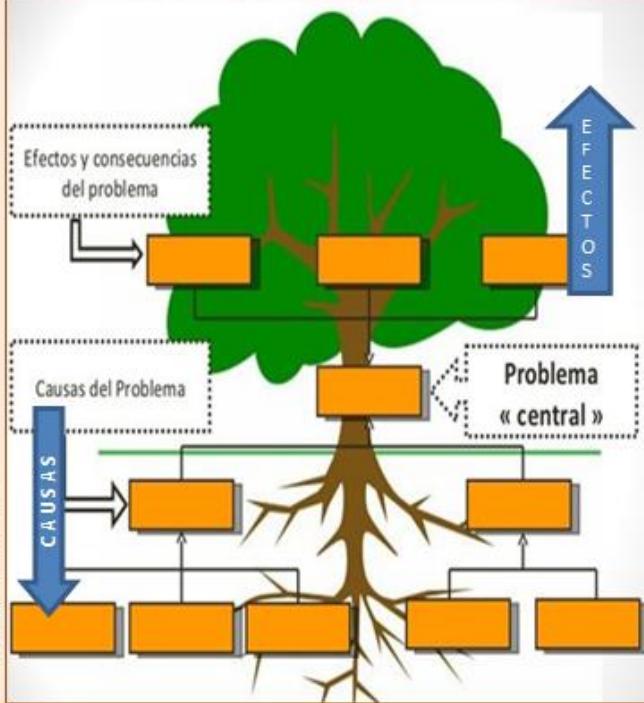


## ÁRBOL DEL PROBLEMA, UNA ALTERNATIVA PARA EL ANALISIS DE SITUACIONES QUE AFECTAN LA CALIDAD DE VIDA EN LAS COMUNIDADES

Palabras claves: lluvia de ideas, involucrados, contexto de la comunidad

- 1.- Seleccione en consenso con los involucrados su **Problema Central** (estado negativo existente).
- 2.- Identificar las causas sustanciales y directas del Problema central.
- 3.- Identificar los efectos sustanciales y directos del Problema central.
- 4.- Construir un árbol de problemas donde de enseñe las relaciones CAUSA-EFECTO.
- 5.- Conciliar con los involucrados SI es VÁLIDO y COMPLETO la red dentro del árbol.

Notas: a.- se puede conducir el análisis del problema, cuando los participantes estén convencidos que allí reposa toda la información esencial que caracteriza el problema . b.-Causas (RAÍCES): son las condiciones que determinan o influyen en la aparición del problema. C.-Efectos (FRUTOS): refieren a las consecuencias e impactos producidas por el problema.



**Imagen:** Contexto teórico del Árbol del Problema. Fuente: Pizarrón Didáctico, elaborada por Canela, M (2020).



El árbol del problema en el contexto de **PST**, permite al equipo de trabajo, diseñarlo con la participación de los usuarios, determinando la problemática central, sus causas y efectos.



**Identifica la Problemática en la comunidad seleccionada** con apoyo de los representantes de la comunidad, compañeros de equipo; **diseña el árbol del problema** y entrega a tu profesor asesor en el primer avance del informe.

**ARBOL DE OBJETIVOS:** Según Mairal, D (2015), “un árbol de objetivos es un diagrama utilizado para definir criterios de evaluación de las distintas soluciones a un problema. Se construye a partir de la formulación de del árbol de problemas, resume los objetivos de un actor en concreto y los inconvenientes que impiden que los cumplan”. Sobre el diseño del árbol de objetivo, se concretiza la intencionalidad de la solución. En la imagen 6: Contexto Teórico del Árbol de Objetivos.

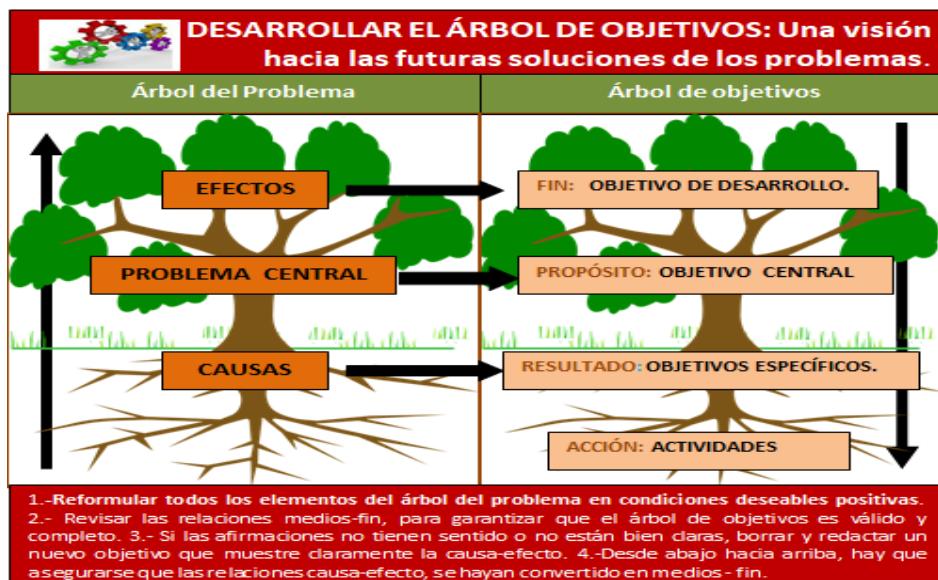
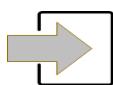


Imagen 6: Contexto Teórico del Árbol de Objetivos. Fuente Pizarrón Didáctico, elaborada por Canela, M (2020).



Determina el Árbol de Objetivos, en este momento se concreta la solución a la problemática detectada. Deben reunirse con los representantes de la comunidad y verificarlo.



**ESTUDIO DE FACTIBILIDAD:** el estudio de factibilidad determina las posibilidades para el desarrollo de un proyecto. Supone la disposición del

recurso humano y tecnológico en el cumplimiento de sus objetivos. En este contexto, el investigador debe apoyarse en un instrumento para recolectar la información y posterior exponer los resultados. Se debe realizar los siguientes estudios:

**FACTIBILIDAD SOCIAL:** Permite determinar el grado de aceptación y apoyo de los actores sociales en la investigación y desarrollo del producto (software).

**FACTIBILIDAD OPERATIVA:** el estudio se relaciona en analizar si el personal que utilizará el producto (software) tiene las competencias para manejarlo.

**FACTIBILIDAD TECNICA:** este aspecto evalúa si las personas poseen los conocimientos técnicos necesarios para desarrollar el software, se reconoce si la institución tiene la infraestructura para implantar el producto.

**FACTIBILIDAD ECONÓMICA:** se debe realizar un análisis de la relación costo beneficio, donde se pueda identificar que el beneficio supera los costos.

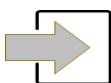
 Sobre este punto, **el equipo del PST**, debe preparar un instrumento (cuestionario), el cual debe ser contestado por los representantes de la comunidad y posterior en función de los resultados obtenidos, pueda continuar con su objeto de estudio o reubicarse en otra organización. Para cada tipo de estudio, debe elaborar varios ítems.



**Realiza el estudio de Factibilidad.** Como equipo de trabajo redacten el instrumento que será contestado por los representantes de la comunidad, insumo necesario para el análisis de resultados.

Ejemplo:

Cuestionario - Estudio de Factibilidad Social			
No.	Ítems	SI	NO
1	¿Considera Ud. que hoy en día la tecnología es importante para mejorar la calidad de vida en el desarrollo de los procesos?		
2	¿Cree Ud. que el uso de una aplicación informática traería beneficios a la empresa?		
3	¿Apoyaría Ud. El desarrollo de una aplicación informática?		
4	¿Estaría Ud. dispuesto a participar en el entrenamiento para el uso de una aplicación informática?		
5	¿Estaría dispuesto en utilizar la aplicación informática en su vida cotidiana laboral?		



### METODOLOGIA DE DESARROLLO DE SOFTWARE (MERINDE):

La Metodología de la Red Nacional de Integración y Desarrollo de Software Libre, es una Propuesta Metodológica para elaborar software Libre con el Uso de Estándares Abiertos, que puede ser empleado y adaptado según los requerimientos de cualquier comunidad u organización. (Imagen 8 ilustra su contexto).



Imagen 8: Contexto de MERINDE. Autor: Canela, M (2023).

MERINDE, mantiene una librería de plantillas reutilizables para la Ingeniería de Software. Estas plantillas proveen un punto partida para los documentos utilizados en proyectos, con lo que pueden ayudar a los desarrolladores a trabajar más rápido y evitar pasar por alto aspectos importantes del proceso de desarrollo. Está concebida para abarcar el

desarrollo completo sea cual sea su complejidad y magnitud, por lo cual su estructura responde a desarrollos de pequeño, mediano y gran alcance.

La metodología está dividida en **cuatro fases**: Fase de **Inicio**, Fase de **Elaboración**, Fase de **Construcción** y Fase de **Transición**; Aunado a un **conjunto de disciplinas** (Modelo del Negocio, Modelo de Requerimientos, Modelo del Análisis y Diseño, Implementación, Pruebas, implantación) (Ver imagen 9: contexto de las fases de MERINDE).

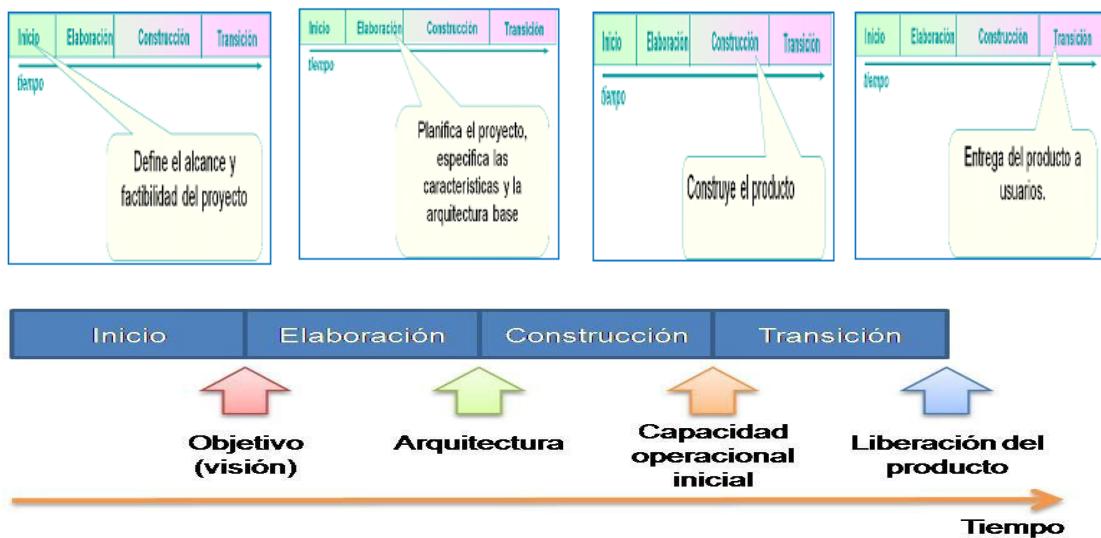


Imagen 9: contexto de las fases de MERINDE: Fuente: Canela, M (2023).

**Las fases se describen como:**

**Fase de Inicio:** se plantea el ciclo de vida del proyecto, se definen los actores, el propósito general y específico del mismo, alcance y casos de uso. Además, se debe desarrollar un plan de negocio para determinar los recursos utilizados a lo largo del proyecto.

**Fase II - Elaboración:** se elabora la arquitectura del ciclo de vida del proyecto, es decir, los requerimientos funcionales y no funcionales, esta fase finaliza cuando se tienen las amenazas y/o riesgos principales que podrían afectar al software, de manera de saber cuáles serían los requerimientos al momento de implementar dicha aplicación.

**Fase III - Construcción:** El objetivo es concluir con el diseño y desarrollar el proyecto en su forma operativa de forma incremental por medio de mejoras a la aplicación, también en esta fase se deben realizar pruebas al proyecto para encontrar posibles debilidades o fallos que pueda contener el mismo.

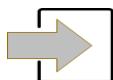
**Fase IV-Transición:** Es la fase final la cual tiene como objetivo específico la entrega del producto a los usuarios finales en su estado funcional y se debe confirmar que los usuarios aprendan a emplear el software de manera eficiente y sin errores para que de esa manera poder avalar que los usuarios estén calificados para dicha aplicación.



**MERINDE**, es la metodología que el grupo de trabajo del PST debe internalizar, ya que, **sobre ella se emprenden las acción en el desarrollo del software**.



**Analiza la fase de inicio y la disciplina modelo del negocio,** para ello lee bibliografía aquí recomendada.



**MODELO DEL NEGOCIO:** Primera disciplina de MERINDE, representa la Arquitectura del Negocio (DAN). En él se utilizan **artefactos UML** para una comprensión universal de los profesionales del área de la informática y para los usuarios. El objeto es exponer la realidad del negocio identificando:

**A.- VISIÓN GENERAL:** describe las metas para la organización e ilustra los problemas que deben ser resueltos para alcanzar esas metas.

Según Darton (1977), los factores importantes a considerar son: Misión, Objetivos Estratégicos, Fortalezas, Debilidades, Oportunidades, Amenazas, Factores Críticos, Unidades organizacionales, Roles y Procesos claves.

**B.-PROCESOS:** presenta el conjunto de acciones mediante el cual los insumos se transforman en un conjunto de resultados valiosos para el cliente. Eje. Cobranza, Inscripción. Los procesos muestran las actividades que deben ser realizadas para alcanzar una meta explícita en el negocio.

**C.-ESTRUCTURA:** Muestra la estructura de los recursos, los productos o los servicios, y la información del negocio, incluyendo la estructura organizacional (divisiones, departamentos, secciones, unidades de negocio, etc.).

**D-COMPORTAMIENTO:** presenta el comportamiento individual de cada recurso y proceso importante en el modelo de negocio.



**La dinámica para PST según esquema normado dentro del PNFI,** el estudiante debe desarrollar una serie de acciones y generar unos artefactos. Aquí se presentan siete (7) pasos:

**1- Ubicar una Comunidad** bien estructurada, es decir, que cuente con un organigrama estructural de sus departamentos y en ello sus procesos.

**2.- Seleccionar el/los departamento(s)** que ameritan mayor intervención social, es decir, existen problemas urgente por corregir desde la aplicación de un software. Debe indicar nombre del departamento y sus objetivos estratégicos tal como se ilustra en el **cuadro 6:** Presentación del Departamento en estudio.

**Cuadro 6:** Presentación del Departamento en estudio

<b>NOMBRE DEPARTAMENTO</b>	<< colocar nombre del departamento >>
<b>OBJETIVOS ESTRATÉGICOS</b>	<< ¿indicar <a href="#">que hace</a> ?, para que lo hace>>

. Fuente Canela, M (2023).

**3.- Diseñar Matriz de Proceso:** es decir, debe indicar los procesos que se desarrollan en el departamento indicando nombre, entradas, breve descripción, y salidas tal como se indica en el siguiente **cuadro 7- Matriz de Procesos del Departamento**.

**Cuadro 7:** Matriz de Procesos.

DEPARTAMENTO SELECCIONADO				
ENTRADA(S)	PROCESO	DESCRIPCIÓN	SALIDA	ACTORE(S)
	<<nombre del proceso 1>>			
	<<nombre del proceso 2>>			

Fuente Canela, M (2023).

**4.- Diseñar el modelo del análisis:** se debe realizar una plantilla para cada proceso tomado como objeto de estudio. En ella se indica: nombre del proceso, trabajadores, Reglas del Negocio, Entidades del Negocio/Descripción y Objetivos Estratégicos. En **cuadro 8 - Plantilla del modelo del análisis** y la imagen 10 se ilustra contexto para la elaboración del modelo del análisis.

**Cuadro 8:** Plantilla del modelo del análisis.

<b>NOMBRE DEL PROCESO</b>	<< indique nombre proceso seleccionado>>
<b>TRABAJADORES (ROLES)</b>	<< indique los roles que asumen el personal>>

<b>ENTIDADES DEL NEGOCIO</b>	<<indique nombre de documentos que acompañan al proceso>>
<b>REGLAS DEL NEGOCIO</b>	<<indique normas /reglas que debe cumplir el proceso antes, durante y para cerrar el proceso>>
<b>BREVE DESCRIPCIÓN</b>	<<indique paso a paso el proceso >>
<b>OBJETIVOS ESTRATEGICOS</b>	<<indique los objetivos respetando la redacción de iniciar con un verbo en infinitivo (que hace) y el para que se hace >>

Fuente Canela, M (2023).

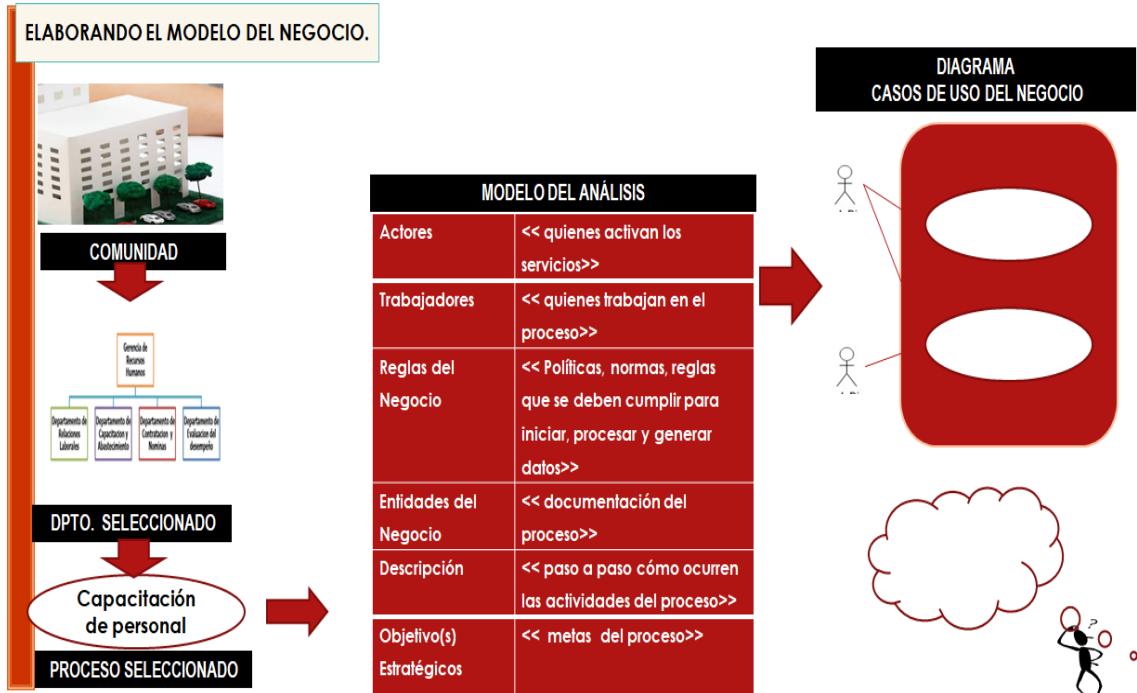
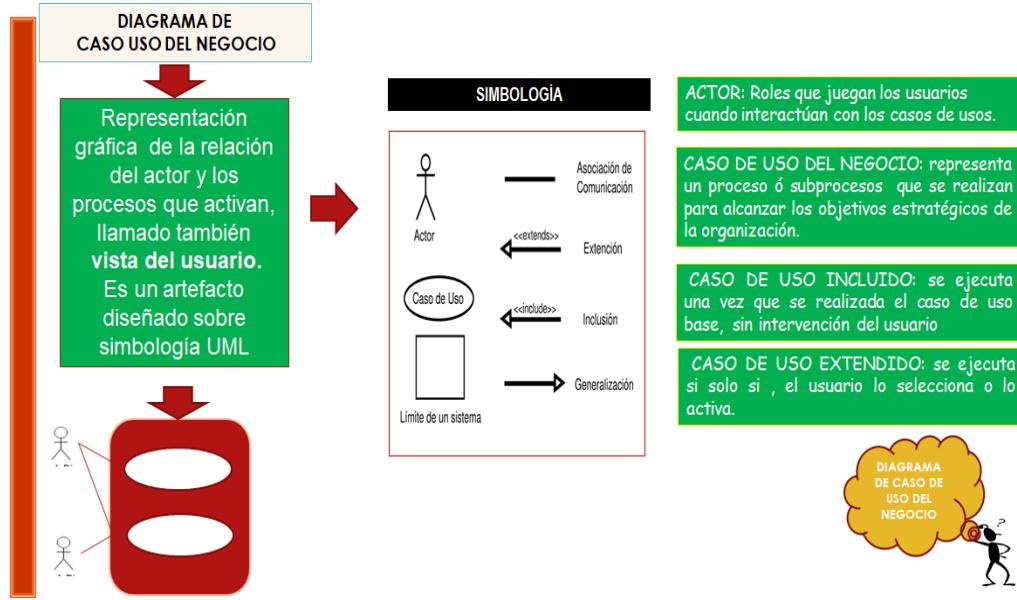


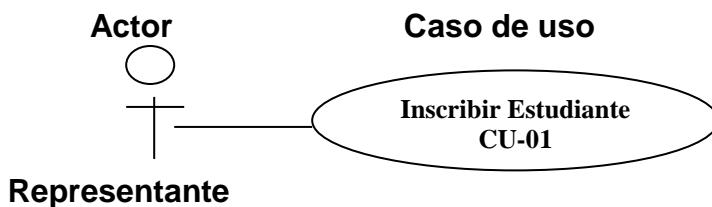
Imagen 10:- Contexto- Elaborando Modelo del Negocio Fuente: Canela (2023).

**5.- Diseñar el Diagrama de Caso de Uso del Negocio:** es la representación gráfica utilizando la simbología UML de los subprocesos que ocurren dentro del proceso seleccionado y se desprenden del ¿Qué? en el objetivo estratégico del modelo del análisis. La **Imagen 11**, ilustra elaborando el diagrama de caso de uso del negocio.



**Imagen 11:** Contexto- Elaborando Diagrama de Caso de Uso del Negocio.  
Fuente: Canela (2023).

**6.- Especificación del escenario de cada caso de uso:** es una plantilla para describir el caso de uso indicando: identificador del caso de uso/ Nombre del caso de uso/ Actor(es)/Propósito/ Precondición/Post condición/ Escenario principal del proceso/ Escenario Secundario del proceso. A continuación plantilla para especificar el caso de uso.



**Plantilla de especificación de un caso de uso**

Identificador del caso de Uso	CU-01	
Caso de uso:	Inscribir Estudiante	Actor(es): Representante
Propósito:		
Precondición:	<<Indicar que condiciones debe cumplirse para iniciar el proceso o caso de uso>>	
Post condición:	<<Indicar que se cumplió y que se obtuvo al finalizar el	

	proceso o caso de uso>>
<b>CURSO NORMAL DE EVENTOS</b>	
<b>Acción Actor</b>	<b>Respuesta del Proceso</b>
1.- Acción 1	2.-Accion 2
	3- Acción 3
	3.1 Si acción 3 es positiva
4.- Acción 4	5.-Accion 5.
<b>CURSOS ALTERNOS DE EVENTOS</b>	
<b>Acción Actor</b>	<b>Respuesta del Proceso</b>
	3.2 Si acción 3 no se cumple
<b>REGLAS DEL NEGOCIO</b>	

**7: Diagrama de Actividad:** es un diagrama UML que permite modelar el comportamiento dinámico de un procedimiento o de un caso de uso; en él se puede expresar de forma gráfica un algoritmo de las acciones que acurren dentro del caso de uso o proceso. El flujo de trabajo va desde un punto inicial hasta un punto final.



En el contexto **de PST**, **cada caso de uso** debe tener asociado **un diagrama de actividad** según la descripción indicada en la plantilla de especificación. A continuación simbología para diseñar diagramas de actividad.

#### Simbología para el Diseño Diagrama de Actividad.

SIMBOLO	SIGNIFICADO
●	<b>Nodo inicial.</b> Indica el punto de partida de las acciones
Nombre	<b>Actividad:</b> el nombre inicia con un verbo y representa una acción
◇	<b>Decisión.</b> Indica las alternativas de una condición.
→	

	<b>Flujo o transición:</b> conexión que dirige el orden a lo largo de los nodos de actividad
—	<b>Concurrencia:</b> indica el inicio o fin de acciones que se realizan concurrentemente.
●	<b>Nodo Final:</b> Indica el punto final de las acciones de un recorrido

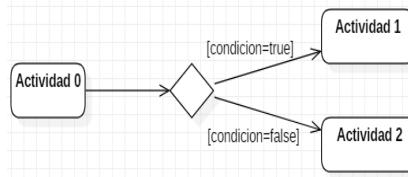
Flujo entre actividades:



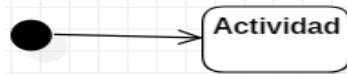
Notación de un flujo de actividad

:

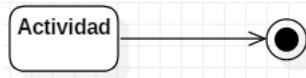
Los flujos de actividades **pueden usar condiciones para su actuación**, estas condiciones **se representan mediante un rombo**, con la condición escrita entre corchetes



Los **nodos iniciales** se muestran como un pequeño círculo relleno.

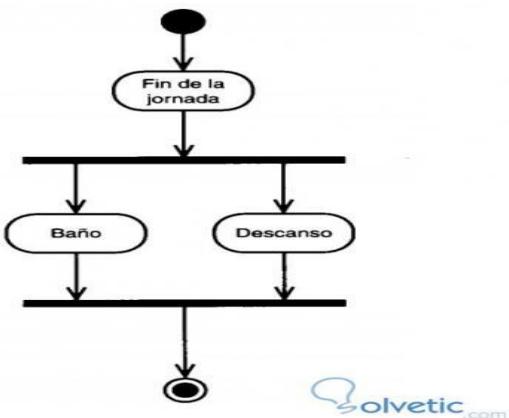


Los **nodos finales** de actividad se muestran como un círculo sólido con un

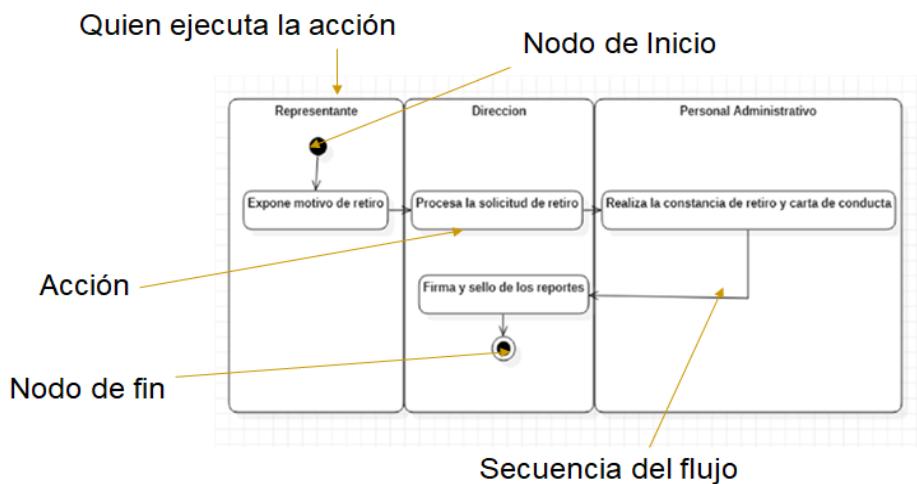


círculo hueco dentro:

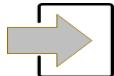
Las **rutas recurrentes** las representamos con una línea horizontal más gruesa y de ella derivan las flechas que llevan a las actividades a ejecutar, luego convergen en otra línea gruesa como la que dio inicio a la ruta recurrente y lleva a un camino en común.



**Diagrama de Actividad**



**Identifica los elementos que constituyen el modelo del negocio.** Uds. como equipo de proyecto, deben socializar los consejos prácticos que se señalan aquí, ya que ellos orientan el desarrollo de las actividades por emprender. **Entrega Borrador Avance del Informe.**



**LENGUAJE UNIFICADO DE MODELADO (UML):** El Lenguaje Unificado de Modelado (UML): se ha convertido en el lenguaje aceptado universalmente para modelar el desarrollo del software; ilustra desde diagramas las ideas, las cuales pueden ser reutilizadas. Los mismos dan nombre y forma a heurísticas abstractas, reglas y buenas técnicas orientadas a objeto. Los tipos de diagramas que contempla son: **Diagrama de Caso de Uso** (útil en el modelo del negocio y modelo del sistema), **Diagrama de Actividad**, Diagrama de Clase: Diagrama de Secuencia, Diagrama de Componente y Diagrama de Despliegue.



Ahora bien, en el contexto del Modelo del Negocio, de UML solo **se pone en práctica los diagramas de caso del negocio y diagrama de actividad para los casos de uso**. Ellos contribuyen en presentar gráficamente el comportamiento del modelo del negocio.



**EVALUACION:** Indica en el paréntesis la letra de la opción correcta:

1.- **La planificación de proyecto permite:** (\_\_\_\_)

- a.- Orientar acciones para elaborar el modelo del análisis.
- b.- Evaluar los avances del proyecto.
- c.- Asignar responsables en las fases de un proyecto.
- d.- Ninguna de las anteriores.

2.- **El equipo de trabajo lo constituye:** (\_\_\_\_)

- a.- Solo estudiantes.
- b.- Estudiantes y Docente Asesor.
- c.- Estudiante, comunidad y docente asesor.
- d.- Ninguna de las anteriores

3.- **El Árbol del Problema es un artefacto de la metodología** (\_\_\_\_)

- a.- MERINDE
- b.- UML
- c.- Marco Lógico.
- d.- Ninguna de las anteriores.

4.- **El Diagrama de caso de uso es un artefacto de la metodología** (\_\_\_\_)

- a.- MERINDE
- b.- UML
- c.- Marco Lógico.
- d.- Ninguna de las anteriores.

5.- **El Diagrama de Actividad permite** (\_\_\_\_)

- a.- Describir el equipo de proyecto.
- b.- Describir el problema
- c.- Describir un proceso de forma gráfica.
- d.- Ninguna de las anteriores.

6.-**El proyecto socio tecnológico permite** (\_\_\_\_)

- a.-Aprender haciendo.
- b.-Tener Práctica Profesional.
- c.-Dar soluciones a una comunidad
- d.-Todas las anteriores.

7.- **El modelo del negocio permite** (\_\_\_\_)

- a.- Conocer el contexto de una comunidad.
- b.- Identificar el objetivo del proyecto.
- c.- Poner en práctica la notación UML
- d.- Ninguna de las anteriores.

**8.- Quién elabora los diagramas (\_\_\_\_)**

- a. Programador.
- b.- Probador
- c.- Líder del proyecto
- d.- Ninguna de las anteriores.



**NOTAS**


# **PRIMERA PARTE**

## **ELABORANDO EL MODELO DEL NEGOCIO**

### **ACTIVIDADES**

#### **ASESORIAS**



Las asesorías son encuentros para aclarar dudas sobre los temas y respeto a concretar el avance proyecto. El profesor asesor las planifica, ya que, pueden ser por grupo o para toda la sección de la UC-PST.

#### **LECTURAS**



El estudiante de PST debe apropiarse del contexto de MERINDE y UML, por lo tanto, debe realizar lecturas constantes a la medida que avanza el proyecto.

#### **FICHA EQUIPO DE TRABAJO**



El equipo, una vez **concretado la comunidad, alcance del proyecto**, el líder del equipo de trabajo debe llenar la ficha con los dato del proyecto. (Ver anexo 1)

#### **INFOGRAFÍA – FASE I - PLANIFICACIÓN DEL PROYECTO**



El equipo de PST, una vez concluida la **FASE I – PLANIFICACIÓN DEL PROYECTO**, debe preparar la infografía de la fase, como material de apoyo para el punto de control.

#### **AVANCE INFORME**



El equipo de PST, debe tomar el esquema que corresponde a su trayecto, y desarrollar paso a paso la **FASE I – PLANIFICACIÓN DEL PROYECTO**, la cual debe entregar como indica el profesor asesor.



## Elaborar la **FICHA DEL EQUIPO DE TRABAJO DE PROYECTO.**

LAPSO ----- UNIDAD CURRICULAR PST \_\_\_\_ SECCIÓN ---- PROFESOR: \_\_\_\_\_  
Correo de enlace con estudiantes:

### FICHA DE PRESENTACION PROYECTO

#### DATOS COMUNIDAD

NOMBRE  
DIRECCION  
TELEFONO(S)

#### DATOS DE LA PERSONA DE ENALCE CON LA COMUNIDAD

NOMBRE  
CARGO:  
TELEFONO(S):

CÉDULA:  
DEPARTAMENTO:  
CORREO:

#### DATOS DEL TUTOR ACADÉMICO

NOMBRE  
TELEFONO

CÉDULA  
CORREO

#### NOMBRE DEL SOFTWARE

#### ALCANCE DEL SOFTWARE

MODULO A:  
MODULO B:  
MODULO C:  
MODULO D: SEGURIDAD

#### DATOS DE LOS INTEGRANTES DEL PST III

Nº	APELLIDOS	NOMBRES	C.I.	CORREO	TELEFONO
1					
2					
3					
4					
5					

# **PRIMERA PARTE**

## **ELABORANDO EL MODELO DEL NEGOCIO**

### **EVALUACIÓN**

#### **PUNTO DE CONTROL I – MODELO DEL NEGOCIO.**



**Punto de Control:** es un **encuentro planificado** por el profesor asesor, para **socializar el avance del informe** del PST y con ello **evidenciar la comprensión del objeto del estudio y alcance de la solución**. El equipo debe apoyarse en una **infografía** para su exposición, como material de apoyo que oriente el discurso de todos los integrantes. Participa el equipo de trabajo/docente asesor/jurado; esta actividad de vital importancia **permite concretar algunos faltantes y conciliar las ideas sobre la propuesta**. Se debe realizar en las instalaciones de la universidad.

#### **AVANCE DEL INFORME – MODELO DEL NEGOCIO**



**El avance del informe:** se centra en entregar el desarrollo de la **Fase I – Planificación del Proyecto**, la cual se describe en esquemas 2018 del PNFI. Cuando se entrega, ya se ha definido: **1.- Equipo de trabajo**, **2.-** Se tiene información clara del contexto de la **Comunidad**, **3.-Se tiene el Departamento** seleccionado con representantes de la comunidad, **4.-** Comprensión clara de los **procesos** del departamento, **5.-Modelo del Análisis** de los procesos a mejorar/Diagrama de Caso de Uso del Negocio/Diagramas de Actividad **6.-Problematica** identificada, **7- El Estudio de Factibilidad** que indique que es completamente viable desarrollar e implantar el software, **8.- Clara la alternativa de Solución** según las

exigencias de PST, en este sentido ello establece el alcance del software y estructura tecnológica que cumplir.



**REALIZA EL AVANCE DEL INFORME:** es importante la participación de todos los integrantes del equipo, deben tener la comprensión del todo.

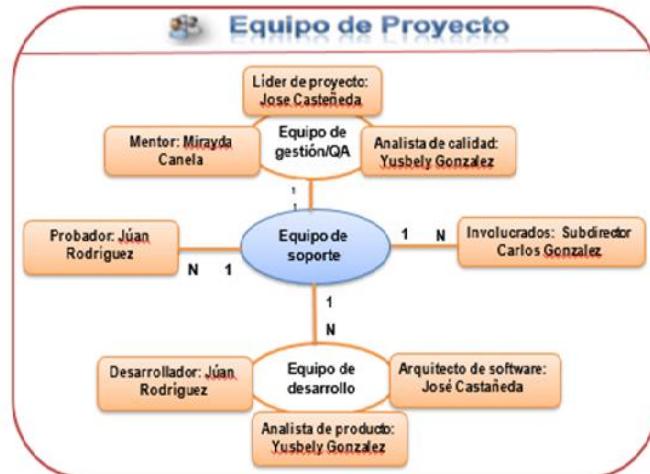


Para la **FASE I – PLANIFICACIÓN DEL PROYECTO**, contempla lo siguiente.

**I. PLANIFICACIÓN DEL PROYECTO:** << Consiste en señalar que es planificación de proyecto citando un autor y dar su postura relacionando su importancia para su trabajo. En este orden de ideas, el objetivo central es la definición y planificación del proyecto para alcanzar los objetivos esperados >>

**Conformación de equipo:** << Consiste en presentar en forma gráfica y descriptiva según MERINDE cómo está estructurado el equipo de trabajo, se debe indicar roles y responsabilidades que serán asumidos por los integrantes, aquí se ilustre ejemplo >>

Organograma estructural del equipo



**Planeación de las fases a abordar para la ejecución del Proyecto:** << Consiste en indicar paso a paso las actividades a realizar para completar el proyecto; las mismas deben orientarse por las fases y disciplinas de MERINDE, aunado a los artefactos UML que se diseñan a lo largo del desarrollo del PST, se expresa como cronograma de actividades del proyecto, aquí un ejemplo >>

### Cronograma de actividades del proyecto.

FASE I - INICIO					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS
Conformar equipo de trabajo					
Abordaje a la Comunidad					
Identificar el problemática					
Estudio de Factibilidad					
Modelo del Negocio					
FASE II - ELABORACIÓN					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS
FASE III - CONSTRUCCIÓN					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS
FASE IV - TRANSICIÓN					
ACTIVIDADES	FECHA INICIO	FECHA FIN	RESULTADO ESPERADO	RESPONSABLE(S)	RECURSOS

.Fuente: Canela, M (2023).

**Alcance del Proyecto:** << Aquí de forma descriptiva indicar que contempla su proyecto. Recuerde que uno de los principales propósitos del PST, es contribuir con la soberanía tecnológica, y promover el uso del software para mejorar los procesos en las comunidades. En esta oportunidad el grupo **propone un software** y debe indicar operaciones podrá realizar el usuario con ello>>

## II. DIAGNÓSTICO SITUACIONAL PARTICIPATIVO Y COMUNITARIO.

**Abordaje Exploratorio Técnico Comunitario:** En la investigación de la ciencia social es indispensable la incursión del investigador en el campo de estudio o en la población objeto de estudio. Para esto se utilizan técnicas de investigación específica, acorde y apropiada para un abordaje efectivo. En este sentido, **el abordaje se puede entender como “la articulación de respuestas a las necesidades de transformación de una realidad social, a través de la construcción, la creación y el vivir.”** De modo que el abordaje es la conexión de los actores sociales para la búsqueda conjunta de soluciones factibles a una necesidad detectada. En **consecuencia, se hace necesario el acercamiento a la comunidad específica en la cual se desarrollará el presente proyecto**, cuyo propósito es **establecer inicialmente un vínculo** con la unidad social para diagnosticar y describir las situaciones de conflicto afrontadas. En tal sentido, el abordaje a la comunidad es la inserción del investigador a fin de conocer el hecho a investigar como ocurre en la realidad.

**Procedimiento utilizado.** << Aquí debe insertar **paso a paso como desarrolló el abordaje exploratorio** en la comunidad y **que logró obtener para concretar** la vinculación comunidad-comunidad; Aquí un ejemplo>>

Luego de un recorrido por el entorno de la UPTP “Juan de Jesus Montilla”, el equipo se ubicó en la institución XXXX. Previa comunicación se planifica una entrevista con el Director – XXX, a quien se le explica la intencionalidad del PST; luego de una larga conversación se concilia ubicarnos en el Departamento XXXX. Ese mismo día, se acuerda un encuentro con jefe del departamento para iniciar el levantamiento de información sobre el contexto del mismo. En este orden, como manifestaron estar totalmente de acuerdo con la idea planteada, se solicitaron la carta de presentación ante la coordinación del PNFI y formalizar la vinculación universidad-comunidad.

Es importante señalar que, durante **visitas realizadas a la comunidad, se aplicó un guion de preguntas (entrevista) al director XXXX** que permitió conocer los diversos problemas e identificar el más urgente que atender. Igualmente, se realizan preguntas en relación a las **factibilidades económicas, sociales, técnicas y operativas**.

Posteriormente, **se efectuó las observaciones** con el propósito de **recolectar información pertinente de los procesos manuales**. Se utilizó cámaras fotográficas para obtener imágenes de documentos generados en los procesos con la finalidad de poder identificar en ellos los datos y requisitos, así como videos para grabar las acciones administrativas de los recursos educativos con la finalidad de lograr detectar las tareas y los responsables de realizarlas.

**Con base a toda la información recolectada** por las técnicas usadas, **se concluyó que la solución automatizada a la problemática detectada se llamará Software para XXXXXXXX, el cual permita** agilizar de forma eficiente los procesos de xxxxxxxx.

Los pasos realizados para el abordaje comunitario está basado en la metodología IAP, donde en la investigación se utilizaron diversas técnicas como entrevistas, encuestas y observaciones para obtener información relevante de la problemática prioritaria, el cual es xxxxxxxx, por otra parte, en la acción se planteó la solución automatizada de un software administrativo de recursos educativos y en la participación se permitió a la comunidad

acordar la idea de desarrollar el software mencionado y opinar los avances del software para procurar que sean los requisitos solicitados.

## **Técnicas e instrumentos para la recolección de la información.** << Aquí debe definir la técnica según algún autor e indicar qué uso y qué logro; ejemplo >>

Durante el levantamiento de información, el equipo aplicó las siguientes técnicas:

- **Entrevista:** Según Arias F (2006), “*la entrevista xxxxxxxx*”. En este caso, se empleó el instrumento estructurado denominado guía de entrevista para exponer unas series de preguntas planificadas al subdirector, con lo que permitió obtener información del problema tecnológico que considera la comunidad solventarlas de forma automatizada, el cual es la actividad manual de herramientas.
- **Encuesta:** Según Arias F (2006), “*la encuesta xxxxxxxxx*.” En este caso, al subdirector se le entregó un instrumento escrito llamado cuestionario, donde se le pidió responder a los ítems con respuestas alternativas de si o no. De esta forma, se obtuvo resultados favorables que el proyecto es viable para su ejecución en términos sociales, económicos, operativos y técnicos.
- **Observación:** Según Arias, F (2006), “*La observación consiste en XXXXX* ”. En este caso, se utilizaron instrumentos no estructurados como cámaras fotográficas y videos para visualizar las actividades de registro de herramientas e identificar en ella los procesos, requisitos y actores responsables de su ejecución, así como los documentos generados durante tal proceso, permitiendo señalar los campos necesarios para adaptarlas al sistema.
- En resumen, los instrumentos mencionados, permitieron obtener información relevante para el desarrollo del proyecto tales como la situación problemática de la falta de control de herramientas con lo que surgió la idea que permitiera su solución al agregar el módulo de herramientas a fin de dar continuidad al software construido en el trayecto II y III bajo el nombre de software administrativo de recursos educativos, igualmente de la factibilidad a la hora de asegurar la realización efectiva del mencionado proyecto en todos los aspectos ya sea económico, sociales, operativos y técnicos. Así como de los procesos manuales del mismo junto con los documentos y requisitos empleados durante la nombrada gestión que permitió definir el nuevo requerimiento de herramientas.

## **Factibilidad social, económica, técnica, operativa.** << Aquí debe indicar como realizo el estudio de factibilidad; diseñar cuatro guiones de preguntas, uno por tipo de factibilidad, luego realiza análisis de datos, pudiendo graficar: ejemplo>>

Se utilizó la técnica de la encuesta mediante el uso del cuestionario con preguntas cerradas, con respuesta alternativa de si o no; el resultado esperado arrojó que ciertamente el software se puede instalar e implantar en las instalaciones de la comunidad. A continuación, se muestra la evidencia del estudio:

Factibilidades			
Social	Económico	Técnico	Operativo
Cuenta con la aceptación del subdirector para la ejecución del proyecto, así como el acuerdo del subdirector para permitir al personal manejar la aplicación informática. Además, de su acuerdo en participar activamente en el entrenamiento para el uso de la aplicación informática.	Cumple con los requisitos económicos de reducir el costo del uso del papel para imprimir planillas relacionadas a préstamo y devoluciones de materiales y libros. Además, se cuenta con la disposición de la comunidad para costear el transporte con la finalidad de trasladarse a la universidad para la evaluación de la aplicación informática.	Definitivamente el proyecto puede realizarse sin ningún problema porque cuenta con todos los recursos en buenas condiciones para su puesta en marcha tales como computadoras que tiene los archivos necesarios para ejecutar el manejador de base de datos que permite la operatividad del sistema, también de servidores, y redes locales. También, todas las computadoras tienen instalado el software libre.	Tiene la disponibilidad de recursos humanos con conocimientos informáticos dispuestos a dar soporte a los usuarios en el manejo del sistema. También cuenta con la disposición del personal de participar en el entrenamiento para manejar el sistema, así como en el proceso de implantación para notificar los errores durante su uso, permitiendo la garantía de un producto confiable.

Fuente Castañeda, J (2022).

**Descripción del Contexto de la Comunidad:** << Aquí debe insertar la información que describe la comunidad tal como: **Nombre, Ubicación, Misión, Visión, Reseña Histórica, Objetivos Estratégicos, Organigrama Estructural >>.**

**Intencionalidad o Propósito (General y Específicos):** << Aquí debe señalar el **Objetivo General** del proyecto y los **Objetivos Específicos** del mismo: Aquí un ejemplo:

**Objetivo General:** Aplicar un software para asistir las actividades escolares en la Unidad Educativa Nacional Bolivariana "Santiago Mariño", Acarigua Edo Portuguesa.

#### **Objetivos Específicos:**

Diagnosticar las necesidades de aplicar un software para asistir las actividades escolares en la Unidad Educativa Nacional Bolivariana "Santiago Mariño".

Determinar los requerimientos para aplicar un software que permita asistir las actividades escolares en la Unidad Educativa Nacional Bolivariana "Santiago Mariño".

Diseñar un software para asistir las actividades escolares en la Unidad Educativa Nacional Bolivariana "Santiago Mariño".

Implementar el software para asistir las actividades escolares en la Unidad Educativa Nacional Bolivariana "Santiago Mariño".

Implantar el software para asistir las actividades escolares en la Unidad Educativa Nacional Bolivariana "Santiago Mariño"

**Fundamentación legal:** << Aquí debe señalar algunos artículos, donde respalde la legalidad del uso de la tecnología en los procesos

administrativos para mejorar su funcionamiento, relacione porque. Debe iniciar con la Constitución de la República Bolivariana de Venezuela, continuar con otros tales como Ley Plan de la Patria, gacetas, decretos, entre otros >>.

**III. MODELO DEL NEGOCIO:** <<Es una descripción de los procesos de una organización y este proporcionara la comprensión del negocio del cliente como un todo. Con este conocimiento es posible recomendar al cliente que le puede ofrecer el software para que el mismo desde el uso de la tecnología puede mejorar los procesos. El modelaje se realiza utilizando UML y la disciplina está indicada en MERINDE. >>

**3.1-UNIDADES EN ESTUDIOS:** << se centra en describir al/los departamentos seleccionados en conciliación con el usuario durante el abordaje. Si son más de un departamento, presentar el estudio uno por uno para ello debe indicar:>>

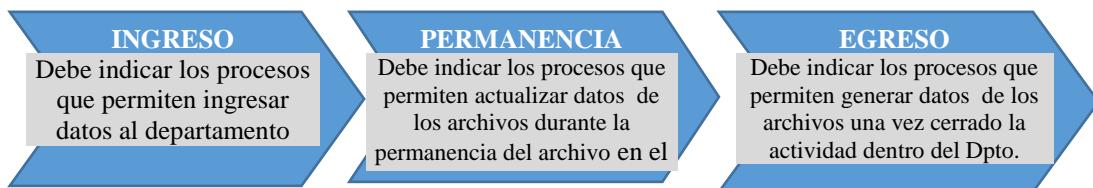
**Nombre (s):** << debe señalar claramente nombre del Departamento:>>

**Objetivos Estratégicos:** << debe señalar claramente los objetivos indicando que hace y para que se hace Departamento: Ejemplo: Realizar la **inscripción del estudiante**, para ubicarlo formalmente en la nómina estudiantil de la institución >>

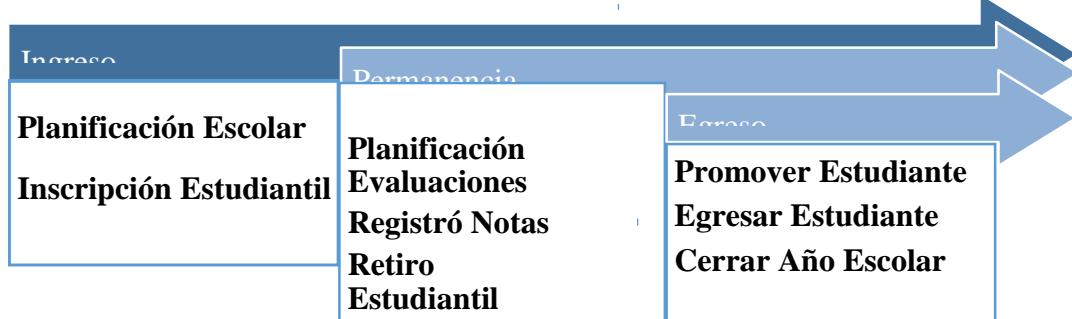
**Especificación de los Procesos:** >><< debe señalar y describir cada proceso que se realiza dentro del departamento, para ello es fundamental establecer la relación de que un objetivo estratégico demanda un proceso; utilice el cuadro indicado en este contexto para tal fin: ejemplo

ENTRADAS	PROCESO	DESCRIPCIÓN	SALIDA	ACTORES
Datos Personales + Grado a Cursar	<b>Realizar Inscripción Estudiante</b>	El representante solicita incripción del niño..... ..... .....	<ul style="list-style-type: none"><li>• Constancia Inscripción</li><li>• Nomina Estudiante Actualizada</li></ul>	Representante
Datos del estudiante + Notas por corte	<b>Registrar Notas del estudiante</b>	El docente toma su listado de clase y registra las notas obtenidas por estudiante, la contabiliza y registra en boletín de cote. Finalizado el año escolar, promedia y genera boletín final	Boletín por Corte  Boletín Final	Docente

**Ficha o Mapa de Procesos:** << en este punto debe presentar gráficamente la relación de los procesos dentro del departamento en una estructura de INGRESO/PERMANENCIA/EGRESO: aquí se señalan ejemplos>>



### EJEMPLO



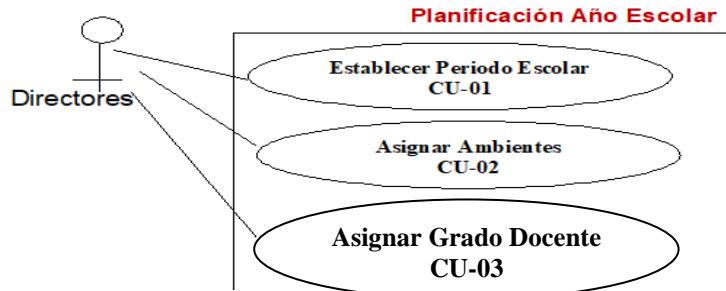
**3.2.-NECESIDAD TECNOLÓGICA:** << se centra en describir detalladamente el/los procesos seleccionados que será atendida la problemática identificada. **Nota: por cada proceso seleccionado se le debe realizar una plantilla del modelo del análisis>>**

**Modelo de Análisis:** << Aquí debe insertar la plantilla para tal fin, es importante reconocer todos los detalles, ya que serán útiles para modelar el software. Ejemplo>>

PROCESO	Realizar la Planificación del año escolar.
TRABAJADORES (ROLES)	Director(a) / Subdirector(a) / Personal administrativo
ENTIDAD DEL NEGOCIO	<ul style="list-style-type: none"> <li>• Nómina de docentes. / Listado de ambientes /</li> <li>• Listado de estudiantes regulares por grado y sección</li> </ul>
REGLAS DEL NEGOCIO	<ul style="list-style-type: none"> <li>• El periodo anterior debe haber finalizado.</li> <li>• No pueden haber más de un periodo escolar activo.</li> <li>• La matrícula estudiantil debe de estar llena.</li> <li>• Todos los profesores deben de tener asignado su grado con su respectiva sección y ambiente.</li> <li>• La sección se apertura con un mínimo de 15 estudiantes</li> <li>• El año escolar debe estar conformado por 3 lapsos</li> <li>• El año escolar comienza en la fecha que dicta el Ministerio de Educación.</li> </ul>
BREVE DESCRIPCIÓN	El director junto con la subdirectora y el personal administrativo planean el año escolar y discuten la nueva planificación. Tomando en cuenta la opinión de los docentes, esto se hace mediante reuniones administrativas que se va realizando antes de iniciar el año escolar. Al finalizar se tiene lista la planificación para iniciar la inscripción

<b>OBJETIVO ESTRATEGICOS</b>	<ul style="list-style-type: none"> <li><b>Establecer periodo escolar</b>, para tener un registro de todos los estudiantes inscritos en ese periodo.</li> <li><b>Asignar</b> a las secciones sus respectivos <b>ambientes</b>, para que cada sección tenga su ambiente fijo.</li> <li><b>Asignar a los profesores</b> sus respectivos <b>grados</b>, para tener un registro de todos los docentes.</li> </ul>
------------------------------	--

**Diagrama de Caso de Uso del Negocio:** << Aquí debe insertar el diagrama de caso de uso que se desprende del modelo del análisis, específicamente de los objetivos estratégicos. Ejemplo>>

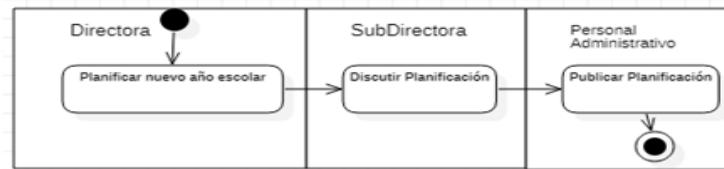


**Plantilla(s) de especificación:** << Aquí debe insertar para cada caso de uso una plantilla es especificación del mismo. Es importante que los flujos principales y alternos se detallen en acciones, las cuales se traducirán posteriormente en un diagrama de actividad. Ejemplo>>

PLANTILLA DE ESPECIFICACIÓN CU-01	
Caso de uso:	Planificar año escolar
Actor(es):	Director(a), Subdirector(a)/Personal administrativo.
Propósito:	Planificar un nuevo año escolar, para luego empezar el registro y asignación de cupos por grado y sección.
Precondición:	Líneamientos del ministerio y la proyección.
Post condición:	Año escolar, grado y secciones con su respectivo docente asignado.
CURSO NORMAL DE EVENTOS	
Acción actor	Respuesta del Proceso Actual
1. La Zona Educativa solicita a la institución planificar el año escolar. 2. La Zona Educativa fija la fecha de inicio del año escolar junto a las fechas de inicio y cierre de cada lapso escolar.	3. El director se reúne con todo el personal administrativo para organizar la planificación del año escolar. 4. Asignar a los docentes presentes sus respectivos grados y secciones. 5. Organizar los ambientes por sección y grado. 6. El director publica la planificación escolar.
CURSOS ALTERNOS DE EVENTOS	
Acción actor	Respuesta del Proceso Actual
REGLAS DEL NEGOCIO	
<ul style="list-style-type: none"> <li>El periodo anterior debe haber finalizado.</li> <li>No pueden haber más de un periodo escolar activo.</li> <li>La matrícula estudiantil debe de estar llena.</li> <li>Todos los profesores deben de tener asignado su grado con su respectiva sección y ambiente.</li> <li>La sección de apertura con un mínimo de 15 estudiantes</li> <li>El año escolar debe estar conformado por 3 lapsos.</li> <li>El año escolar comienza en la fecha que dicta el Ministerio de Educación.</li> <li>Durante la planificación del año escolar el Ministerio de Educación debe informar el tema del Proyecto Educativo Integral Comunitario (PEIC) como eje principal en la planificación de actividades.</li> </ul>	

**Diagrama de Actividad:** << Aquí debe insertar el diagrama de actividad que se desprende del curso normal y alterno de las acciones especificadas

**en la plantilla de especificación del caso de uso;** Nota no es más que un algoritmo en forma gráfica. Ejemplo>>



**Formulación del Problema:** << Aquí debe insertar la matriz FODA / árbol del problema construido con la participación de todos los involucrados y luego describir brevemente el estudio. Ejemplo>>

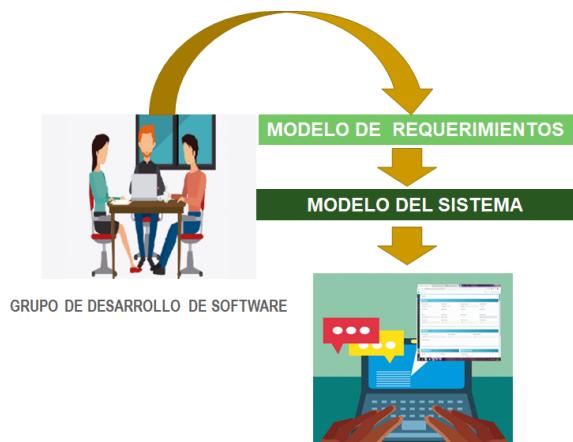


# TERCER COMPONENTE

## CONTENIDO

## SEGUNDA PARTE

### ELABORANDO EL MODELO DEL SISTEMA



*"Es tarea del diseñador, realizar la transición de la funcionalidad a la usabilidad,*

*Lo que significa que el funcionamiento de un sistema computarizado debe corresponder a los requerimientos y habilidades del usuario en su ambiente natural de trabajo" (p365).*

*Narciso, F y Rodríguez, T (2004).*

## DESARROLLANDO EL MODELO DEL SISTEMA

### OBJETIVO(S)



- Determinar los requerimientos funcionales y no funcionales, para concretar el **Modelo de Caso de Uso del Sistema**.

### TEMAS



- 2.1 Requerimientos del Software.
- 2.2 Diagrama de Caso de Uso del Software.
- 2.3 Arquitectura del Software.
- 2.4 Modelo de Base De Dato.
- 2.5 Modelo de Diseño.
- 2.6 Mapa de Navegación.
- 2.7 Interfaz de Usuario.
- 2.8 Diagrama de Componente.
- 2.9 Diagrama de Despliegue.

### ACTIVIDADES



- Asesoría.
- Lectura REQUERIMIENTOS/ UML / MERINDE/
- Infografía - Modelo del Sistema

### EVALUACIÓN



- Punto de Control – Modelo del Sistema
- Entrega Avance del Informe

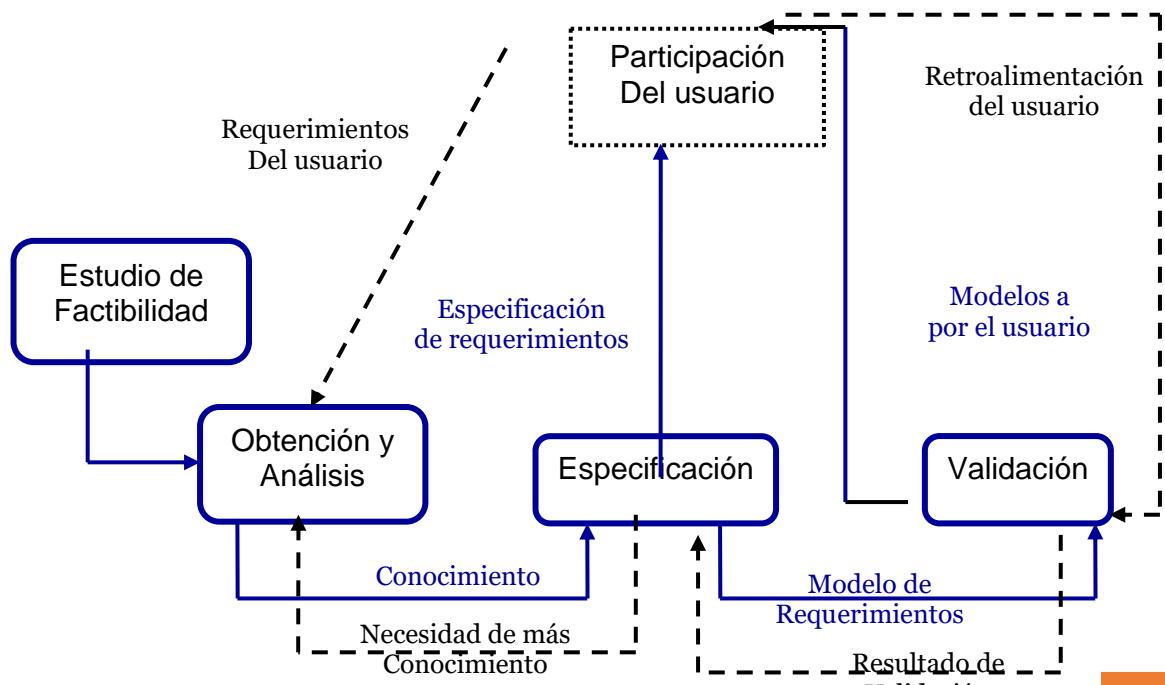
## SEGUNDA PARTE

# ELABORANDO EL MODELO DEL SISTEMA

### TEMAS



**REQUERIMIENTOS DEL SOFTWARE:** al respecto Sommerville, I (2005) expresa: “**son declaraciones en lenguaje natural y en diagramas, de los servicios que se espera que el sistema provea y de las restricciones bajo las cuales debe operar**” (p.46). En este sentido, el analista de sistema, desde una serie de entrevistas con los usuarios, revisión documental y observaciones; debe describir y analizar muy bien los acuerdos establecidos. **La comprensión de los requerimientos, demanda el éxito de las funcionalidades del software;** deben estar **especificado de forma completa sin ambigüedades y posible de probar o verificar;** esto implica reconocer que es la clave para iniciar la construcción del software sobre bases sólidas; es determinar las exigencias del producto y mitigar los riesgos. Se recomienda seguir el **proceso de Ingeniería de Requerimientos como se ilustra a continuación:**

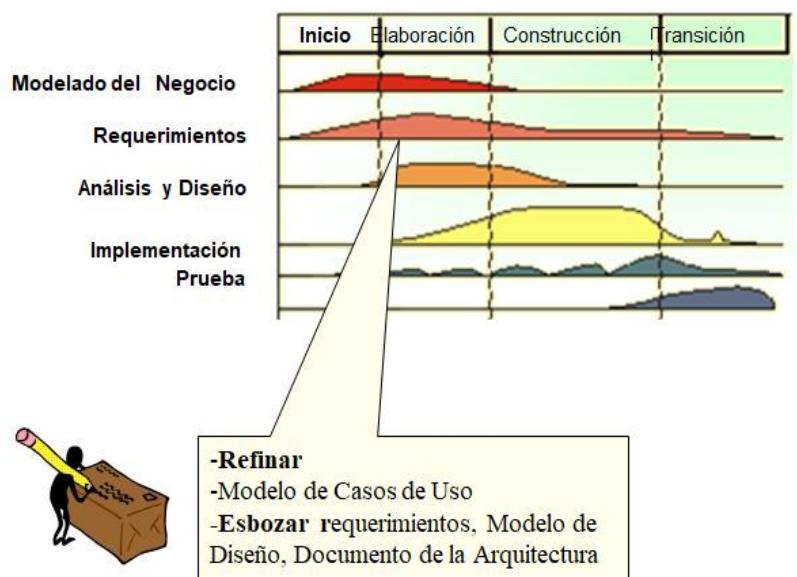




Cabe destacar, que la determinación de **requerimientos**, se encuentra ubicada en **MERINDE** como la **segunda disciplina en el desarrollo de Software**, en la imagen 11 se ilustra; igualmente aclarar que los **requerimientos funcionales describen al software en términos de entradas/procesamiento/salidas**, mientras que los **no-funcionales, en términos de cualidades deseables del sistema** (Interfaz de usuario, Documentación, Hardware, Características de desempeño, Asuntos de calidad, entre otros). **Para identificar los requerimientos se deben utilizar las técnicas e instrumentos de recolección de información.** Debemos recordar que unas de las razones que motivan a las organizaciones abordar la tecnología son hechos como: solucionar un problema en sistemas manuales como automatizados, aprovechar la oportunidad, la competitividad empresarial o una inquietud de directiva. Cada caso exige innovación en la implementación de los requerimientos.

## Disciplinas

## Fases



MÉTODO DE LA RED NACIONAL DE INTEGRACIÓN  
DESARROLLO DE SOFTWARE LIBRE  
MERINDE



Este documento describe las principales fases y el orden de la metodología

Imagen 11: Disciplina Requerimientos. Fuente: Canela, M (2023).



**Identifica los requerimientos** (funcionales y no funcionales) conciliados con el usuario. Ellos van a definir el contexto del software. Utiliza las plantillas **lista de requerimientos funcionales** (cuadro 11), **requerimientos No funcionales** (cuadro 12) y **especificación de cada requerimiento** (cuadro 13).

**Cuadro 11: Plantilla – Lista de Requerimientos Funcionales (RF).**

CÓDIGO	NOMBRE DEL CASO DE USO	ACTORES
RF-01	<< Para cada RF señale el nombre utilizando un verbo en infinitivo y el nombre del objeto, ejemplo <b>Inscribir Estudiante&gt;&gt;</b>	<< señalar quién o quienes, activara la funcionalidad en el software. >>
RF-02		
RF-03		

Fuente: Canela, M (2023).

**Cuadro 12: Plantilla – Especificación Requerimiento Funcional.**

ID Requerimiento:	RF-1
Nombre:	<<Indique nombre del caso de uso>>
Tipo:	<<Indique nombre tipo>>
Descripción:	<< Aquí debe detallar en que consiste el requerimientos>>
Importancia:	<<Indique la importancia: media o alta>>

Fuente: Canela, M (2023).

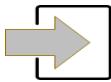
**Cuadro 13: Plantilla – Lista de Requerimientos No Funcionales (RNF).**

Tipo	Código	Descripción
Restricciones del diseño	RNF-01	
	RNF-02	
Interfaz de usuario	RNF-03	
Seguridad	RNF-04	
	RNF-05	

Fuente: Canela, M (2023).



Elabora un plan para identificar los requerimientos, describirlos, verificarlos antes de emprender los diagramas.



### **DIAGRAMA DE CASO DE USO DEL SOFTWARE (DCS):**

Ahora bien, existen técnicas y procedimientos **para presentar la captura y análisis de los requerimientos; UML desde el Diagrama de Caso de Uso**, permite ilustrar la vista de los servicios del software según la perspectiva del usuario; igualmente a ellos se les anexa su descripción desde la plantilla de especificación para cada caso de uso. En este contexto, UML es una herramienta esencial para la captura de requerimientos y especificación de los mismos.



En el contexto de **PST**, la **lista de requerimientos** funcionales es el insumo para diseñar el DCS. En el **Imagen 12** – se presenta un ejemplo de Diagrama Caso de Uso del Software, donde los requerimientos funcionales fueron Inscribir Estudiante, Planificación Año Escolar y Egresar Estudiante; los actores la directora y Docente.

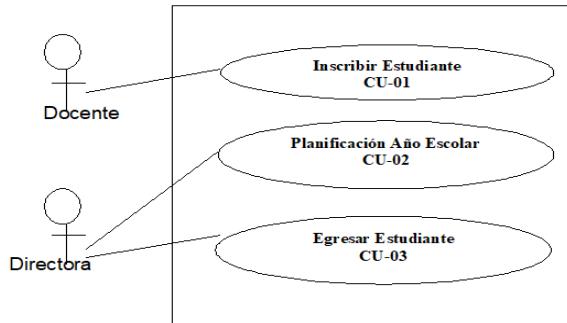


**Nota:** el **Actor** es un usuario del sistema, que necesita o usa alguno de los casos de uso. Un usuario puede jugar más de un rol. Un solo actor puede actuar en muchos casos de uso; recíprocamente, un caso de uso puede tener varios actores. Los actores no necesitan ser humanos pueden ser sistemas externos que necesitan alguna información del sistema actual.



Finalmente para cerrar este apartado debe realizar para **cada caso de uso su plantilla de especificación**, la cual **exige revisión con el usuario**. NOTA: Un **requerimiento funcional** estará representado por un

**caso de uso en el diagrama**, ya sea un caso de uso base o un caso de uso extendido. Un caso de uso extendido es aquel que se activa utilizando una asociación include o exclude. En el ejemplo solo se expresan casos de uso base.



**Imagen 12:** Diagrama de Caso de Uso del Software. Fuente: Canela, M (2023).

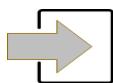
#### Plantilla de especificación de un caso de uso del sistema.

Identificador del caso de Uso		CU-01
Caso de uso:	Inscribir Estudiante	Actor(es): Representante
<b>Propósito:</b>		
<b>Precondición:</b>		Indicar que condiciones debe cumplirse para iniciar el proceso o caso de uso
<b>Post condición:</b>		Indicar que se cumplió al finalizar el proceso o caso de uso
<b>CURSO NORMAL DE EVENTOS</b>		
Acción Actor		Respuesta del Sistema
1.- Acción 1		2.-Accion 2
		3- Acción 3
		3.1 Si acción 3 es positiva
4.- Acción 4		
		5.-Accion 5.
<b>CURSOS ALTERNOS DE EVENTOS</b>		
Acción Actor		Respuesta del Sistema
		3.2 Si acción 3 no se cumple
<b>REGLAS DEL NEGOCIO</b>		

Fuente: Canela, M (2023).



**Elabora el diagrama de caso de uso del sistema y sus plantillas, es recomendable que solicites a tu profesor revisarlas antes de elaborar los diagramas de actividad o de secuencia.**



### **ARQUITECTURA DEL SOFTWARE:** según Plessman, R (2010)

“la arquitectura del software proporciona una visión holística del sistema que se va a construir. Ilustra la estructura y organización de los componentes del software, sus propiedades y conexiones” (p.232). En este sentido, el diseño consistirá en presentar abstracciones coherentes que proporcionen el marco de referencia necesario para guiar la construcción del software.

Esta fase, exige que los analistas, diseñadores y programadores trabajen en una línea común para alcanzar los objetivos. **El diseñador debe lograr la transición de la funcionalidad a la usabilidad del software.** En este escenario, el éxito se centra en **conocer las capacidades y límites de la tecnología, los aspectos sociales y físicos de la organización, los requerimientos funcionales y no funcionales, las capacidades y expectativas de los usuarios**; todos ellos elementos claves para presentar una arquitectura del software adecuada.



El diseñador de la arquitectura para alcanzar la mejor alternativa debe utilizar los requerimientos, los modelos y métodos existentes para tal fin. En la **Imagen 13** la disciplina MERINDE Análisis y Diseño, donde se concreta la arquitectura del software.



En el contexto del Modelo del Sistema, **es la arquitectura del software** la que vincula los artefactos: **Modelo del caso de Uso del Software/ Modelo de Diseño/ Modelo de Datos/ Mapa de Navegación/**

**Interfaz de usuario**, para concretarla y dar inicio a la etapa de la implementación, es decir la construcción del software.

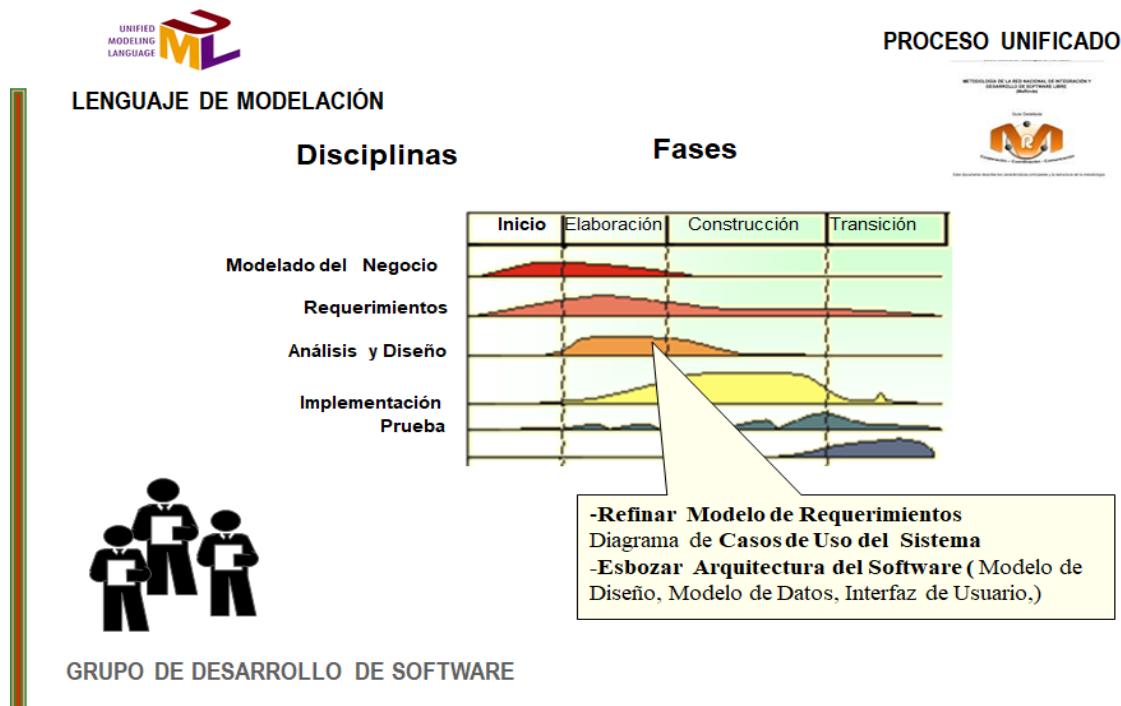
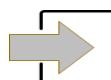


Imagen 22: Disciplina Análisis/Diseño MERINDE. Fuente Canela, M (2023).



**MODELO DE BASE DE DATO:** es una colección de conceptos relacionados que se emplean para describir la estructura de una base de datos. Según Buyens, J (2000): “una base de datos es una serie de datos organizados y relacionados entre sí; existe para dar servicio a las necesidades de las aplicaciones. Una aplicación, a su vez es una unidad de software para dar respuestas a una necesidad pública” (p.19). La **base de dato** en el desarrollo de un software, es el núcleo central para responder a los requerimientos del usuario. Es decir, cada funcionalidad para trabajar necesita un recinto donde reposen los datos de forma organizada y disponible para su uso.



Para diseñar una BD, debe estar ajustado a un paradigma, en este contexto se habla del **Modelo de Datos Relacional**, el cual, en una buena praxis se recomienda elaborar el Modelo conceptual y luego el modelo físico.

**El MODELO CONCEPTUAL** (representa la realidad a un alto nivel de abstracción) el cual exige:

1.- Reconocer y listar las **Entidades** (puede ser una persona, lugar, cosa o evento donde la información es necesaria para el software).

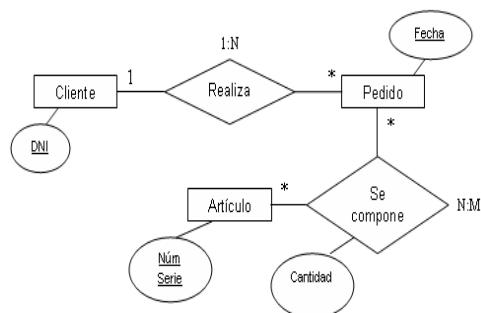
2: Identificar **Atributos** a cada entidad (son las características que definen o identifican a una entidad).

. 3.-Describir las **Relaciones** (es la interacción entre entidades).

4.- Definir la **Cardinalidad** (forma exacta como se relacionan las entidades, es decir, de uno a uno 1:1, de uno a muchos 1: M, o de muchos a muchos M: M).

**El MODELO FISICO** de la base de dato, debe partir del modelo conceptual para crear las tablas en el manejador de base de dato del computador. Ahora el modelo físico refiere tablas con atributos y relacionadas. En el **Imagen 14** se ilustra un ejemplo del modelo de datos.

Modelo conceptual de la BD



Modelo Físico de la BD

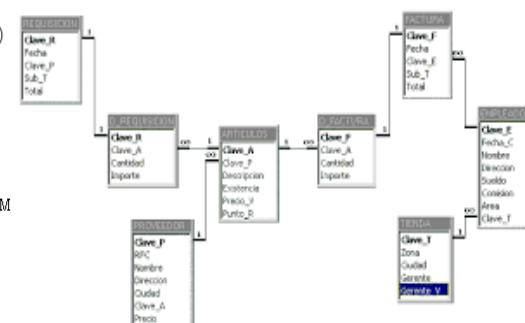
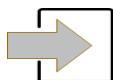


Imagen 14: Ejemplo del modelo de datos. Fuente Canela, M (2023).



Con tus compañeros de equipo, toma los requerimientos y **emprende el diseño de la base de dato**. Es recomendable que la revisen con el profesor asesor o el profesor de base dato para verificar el diseño antes de iniciar el arte de la programación

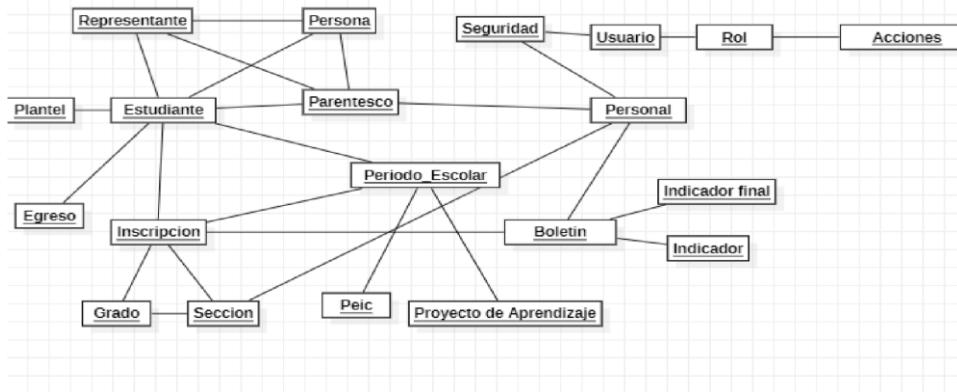


**MODELO DE DISEÑO:** se centra en la definición de la solución mediante la **presentación estructural** de las unidades que permiten comprender mejor el contexto de la programación del software. Resulta del análisis de los requerimientos y pone en marcha el diseño del diagrama de objeto y el diagrama de clase.



**El DIAGRAMA DE OBJETO:** es un diagrama UML que permite describir en un momento determinado la instancia de la estructura estática de un software; **son usados para validar la precisión del diagrama de clases** y comparten virtualmente los mismos símbolos para la notación. Todos los objetos del dominio deben estar relacionados conservando una semántica lógica del contexto.

**Un objeto** es una instancia de una clase. Cada objeto es representado como un rectángulo, que contiene el nombre del objeto, seguido de dos puntos (:) y el nombre de la clase, todo subrayado. **En imagen 15**, se ilustra un **ejemplo de un diagrama de objeto**



**Imagen 15:** Ejemplo de un diagrama de objeto. Fuente Castañeda y otros (2022).



**EL DIAGRAMA DE CLASE:** es un diagrama UML que permite expresar la estructura estática de la información del software. Está compuesto por un conjunto de clases relacionadas entre sí. Recoge los conceptos del dominio de la aplicación, así como los de la implementación.

#### Elementos para diseñar un diagrama de clase:

**Clase** (concepto del dominio del software): es una categoría o grupo de cosas que tienen atributos (propiedades) y acciones similares, es representado por un rectángulo y se divide en tres áreas (nombre de la clase/ atributos /operaciones).

**Relaciones:** permite establecer una conectividad entre clases dando una semántica al contexto; **los tipos son:** **Asociación** (conexión bidireccional entre clases), **Generalización** (mecanismo para crear una super clase), **Agregación** (un objeto compuesto puede agregar otros), **Composición** (un objeto está compuesto por otros). **Dependencia** indica que una clase requiere de otra para proporcionar alguno de sus servicios.

**Multiplicidad:** describe la cardinalidad de la relación, es decir, especifica cuántas instancias de una clase están asociadas a una instancia

de la otra clase. Tipos de multiplicidad: **Uno a uno (1...1)**, **uno a muchos (1...\*)** y **muchos a muchos (\*...\*)**. En la imagen 16, se lustra la estructura de una Clase y en la imagen 17, se ilustra un ejemplo de Diagrama de Clase.

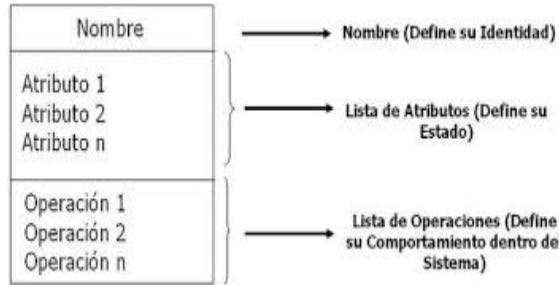


Imagen 16: Estructura de una Clase. Fuente: Canela, M (2023)

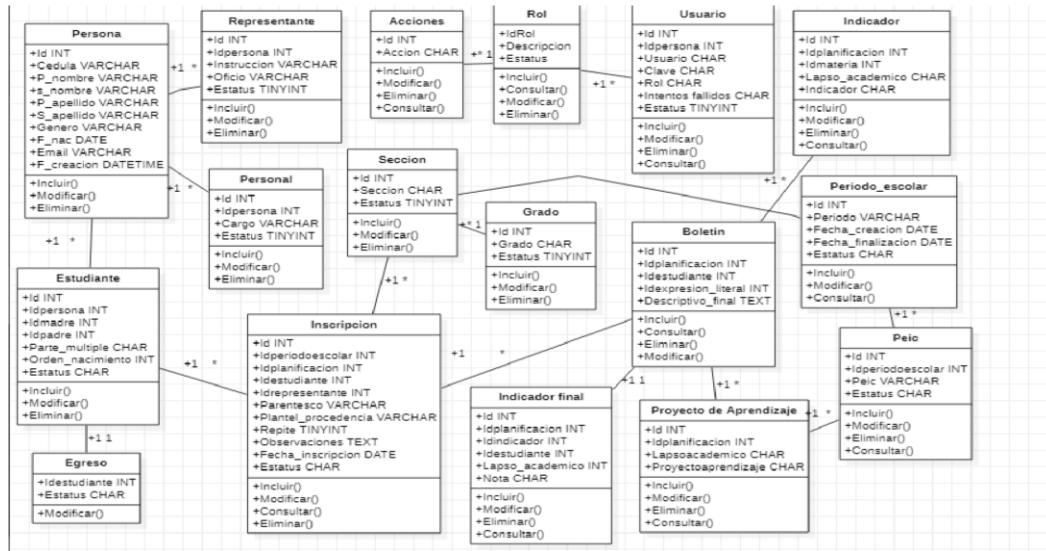
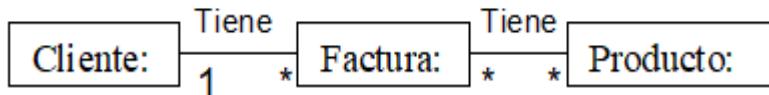


Imagen 17: Ejemplo de Diagrama de Clase Fuente Castañeda y otros (2022),

Ahora bien, **en el contexto del PST, el equipo de trabajo debe tomar los requerimientos funcionales e ir identificando los conceptos candidatos a objetos**, si cada concepto tienen más de dos atributos que lo caracterice será un objeto y luego se convertirá en una Clase. **Ejemplo: En**

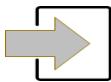
**un contexto de Facturación los conceptos** que se identifican: **Cliente** (Cédula, nombre, teléfono, Dirección), **Producto** (código, nombre, presentación, precio), **Factura** (Número de factura, fecha emisión, forma de pago, monto a cancelar). **Luego se relaciona** y se tiene el diagrama de objeto.



Respetando la estructura de este diagrama, se diseña el diagrama de clase



Con tus compañeros de equipo, toma los requerimientos y **emprende el diseño del diagrama de objeto y el diagrama de clase**. Es recomendable que la revisen con el profesor asesor o el profesor de Ingeniería de Software para verificar el diseño antes de iniciar el arte de la programación.

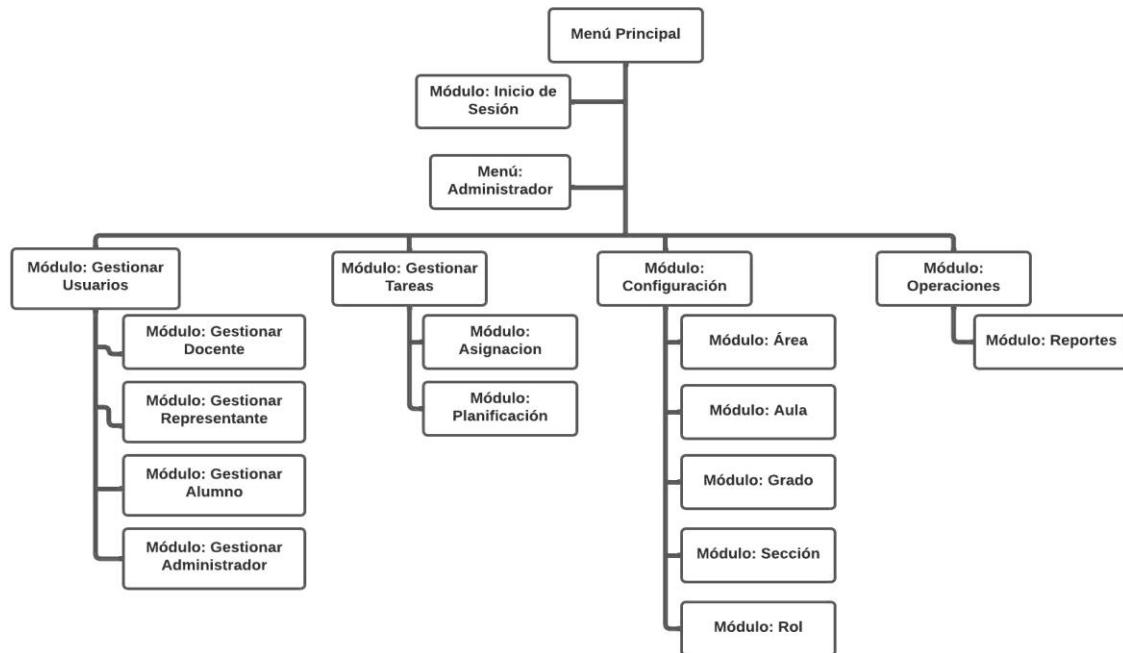


**MAPA DE NAVEGACIÓN:** es un organigrama estructural que permite visualizar las funcionalidades que conforman el software. Su diseño permite crear una cultura de navegabilidad en el usuario de manera ordenada y puntual, así disminuye el tiempo de acceso y respuesta a los servicios incrementando la productividad. La profundidad se gráfica de manera vertical e indica el número de opciones de jerarquía decreciente que hay dentro de una misma sección. En la imagen 18 se ilustra un ejemplo mapa de Navegación.



En el contexto de PST, se recomienda agrupar los formularios por unidades tales como: maestros, transacciones, reportes, ayuda, sin embargo

el diseñador puede expresar otra organización siempre y cuando sea fácil de usar en poco tiempo y poca navegabilidad. En imagen 19 se ilustra un ejemplo.



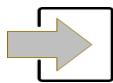
**Imagen 18:** Mapa de Navegación. Fuente Reyes,M (2022).



**Imagen 19:** Ejemplo Mapa de Navegación. Canela, M (2023).



Con tus compañeros de equipo, toma los requerimientos y **emprende el diseño del mapa de Navegación**. Es recomendable que la revisen con el profesor asesor o el profesor de Ingeniería de Software para iniciar el arte de la programación.



**INTERFAZ DE USUARIO:** es la comunicación o diálogo que se desarrolla entre el computador y el usuario. Al respecto Plessman, R (2010) expone “el diseño de la interfaz de usuario crea un **medio eficaz de comunicación entre seres humanos y la computadora**” (p.265). En este orden, **el diseñador** debe seguir los principios para el diseño de interfaz, identificar los objetos, las acciones, y luego **crear una plantilla de pantalla que constituye la base del prototipo** de la interfaz de usuario.



Es importante que **el diseño moldee la percepción que tiene el usuario del software**. El producto final (prototipo) debe ser generar los formatos de las pantallas con los escenarios de las funcionalidades que aspira el usuario utilizar. Debe ser revisado y probado por los usuarios hasta que cumplan todas las expectativas.



Una interfaz de usuario **debe contemplar** símbolos o figuras que representen las **acciones, formularios** con operaciones **de configuración, de transacciones, de consultas, de reportes** para **finalmente evaluar la “Usabilidad”**, atributo de calidad. Cabe destacar que la estructura de cada pantalla se desprende de los formatos usados por el usuario con mejoras de diseño. En **Imagen 19** se ilustra un ejemplo estructura pantalla principal.



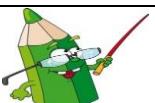
**Imagen 19:** Diseño de Pantalla Principal/ Formularios Maestros. Fuente: Canela, M (2023).

Formularios Maestros, es el diseño de una vista a una tabla de la base de dato que puede funcionar sin problemas

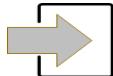
CLIENTE		PRODUCTO	
Cedula	<input type="text"/>	Código	<input type="text"/>
Nombre	<input type="text"/>	Tipo	<input type="text"/>
Teléfono	<input type="text"/>	Precio	<input type="text"/> BS.
Dirección	<input type="text"/>	Nombre	<input type="text"/>
	<input type="button" value="Guardar"/>	Existencia	<input type="text"/>
	<input type="button" value="Cancelar"/>		<input type="button" value="Guardar"/>
	<input type="button" value="Cancelar"/>		



En el escenario de PST, el equipo de trabajo debe diseñar las pantallas de comunicación entre usuario con el computador, las cuales unidas serán la interfaz de usuario. Debe contemplar las operaciones que se pueden realizar sobre una base de dato (incluir, modificar, consultar y eliminar datos según el caso) y que tipo de software va a desarrollar.

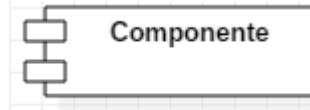


Con tus compañeros de equipo, toma los requerimientos y emprende el diseño de los formularios. Es recomendable que la revisen con el usuario antes de iniciar el arte de la programación.



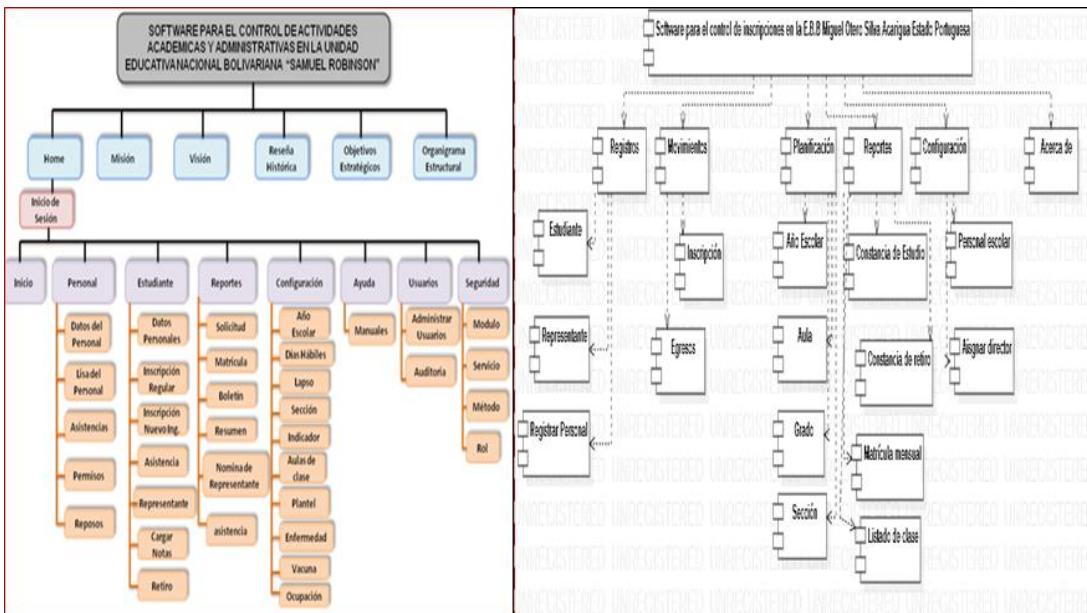
**DIAGRAMA DE COMPONENTE:** el diseño a nivel de componentes se inicia con intención de **traducir el modelo del diseño en un software operacional**. En esta etapa se define las estructuras físicas de datos, algoritmos, características de la interfaz y mecanismos de comunicación para cada componente del software. **Desde el punto de vista orientado a objeto, es un conjunto de clases que se interrelacionan entre sí.**

Es un diagrama UML definido como una parte modular, desplegable y sustituible de un sistema; es un bloque de construcción de software de computo. Un componente **representa una parte modular de un sistema basado en computadora**. Está clasificado como un diagrama de estructura y, como tal, representa la forma estática. El comportamiento se define en términos de interfaces obligatorias y proporcionadas. Los componentes forman parte de la arquitectura del software, y por ende juega un papel fundamental en el logro de los objetivos y requerimientos del sistema. El símbolo de componente es:



**En el caso de PST,** se debe presentar el mapa de navegación en función de la simbología UML. **El mismo se orienta desde el diseño de la carta estructurada.** La estructura debe presentar la organización de las funcionalidades en forma de jerarquía. Muy particular debe tener los **nodos principales** como lo son **el acceso a formularios de configuración, formularios de transacciones y formularios para activar consultas y reportes** entre otros. Aquí un ejemplo:

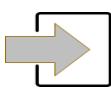
Ejemplo Carta Estructura del Software y Diagrama de Componente del Software



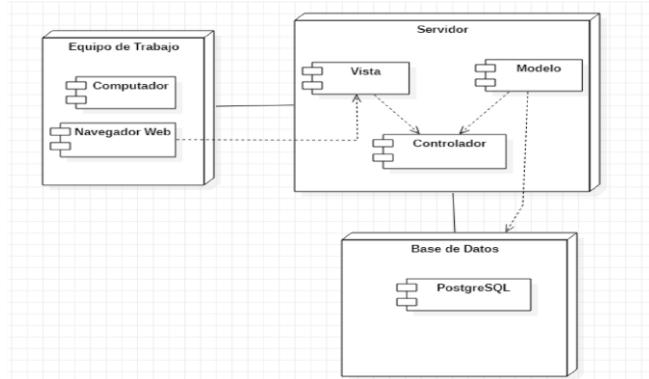
Fuente Canela, M (2023)



**Realiza el diagrama de componente.** Con tus compañeros de equipo, toma la carta estructura y realiza la conversión utilizando la simbología UML para tal fin. Consulta con tu profesor asesor o profesor de ingeniería de software.



**DIAGRAMA DE DESPLIEGUE:** es un diagrama UML, donde se expone la relación entre los componentes de hardware y software. Permite razonar sobre la topología de procesadores y dispositivos sobre los que se ejecutara el software. Los diagramas de despliegue se utilizan para visualizar los aspectos estáticos de estos nodos físicos y sus relaciones y para especificar sus detalles para la construcción, como se muestra a continuación un ejemplo:



**Figura 20:** Ejemplo Diagrama de Despliegue del Software.

Fuente: Reyes, M (2022).



En el contexto de PST, el equipo de trabajo debe señalar la plataforma estructural sobre la cual funcionara el software.



**Realiza el diagrama de despliegue.** Con tus compañeros de equipo, realiza los componentes para que sean señalados en el diagrama



**EVALUACION:** Indica en el paréntesis la letra de la opción correcta:

1.- **Un requerimiento funcional es:** (\_\_\_\_)

- a.- Un caso de uso del software.
- b.- Un caso de uso del modelo del negocio.
- c.- Una funcionalidad del software
- d.- Ninguna de las anteriores.

2.- **El Diagrama de caso de uso del software:** (\_\_\_\_)

- a.- Expresa las funcionalidades del software.
- b.- Expresa la estructura del software.
- c.- Es un artefacto del Marco Lógico.
- d.- Ninguna de las anteriores

3.- **La Base de Dato en el software** (\_\_\_\_)

- a.- Representa la estructura de comportamiento de los datos.
- b.- Es un diagrama UML.
- c.- Es la estructura para la gestión de datos de un software.
- d.- Ninguna de las anteriores.

4.- **El Diagrama de Clase** (\_\_\_\_)

- a.- Es un artefacto de UML
- b.- Es un artefacto de MERINDE
- c.- Es un artefacto del Marco Lógico.
- d.- Ninguna de las anteriores.

5.- **Una clase la constituye** (\_\_\_\_)

- a.- Atributos y métodos.
- b.- Métodos y datos
- c.- Nombre de la clase, atributos y métodos.
- d.- Ninguna de las anteriores.

6.- **Interfaz de usuario** (\_\_\_\_)

- A.-Conjunto de Formularios para la comunicación Usuario-Computador.
- b. Conjunto de Clases que representan el dominio del software.
- C.-Forma de comunicación del personal de la comunidad.
- d.-Todas las anteriores.

7.- **El modelo del sistema permite** (\_\_\_\_)

- a.- Presentar la arquitectura del software.

- b.- Presentar el dominio del software.
  - c.- Presentar el diseño del software.
  - d.- Ninguna de las anteriores.
- 8.- **La carta estructurada orienta (\_\_\_\_)**
- a. Al diagrama de componente.
  - b.- Al diagrama de Despliegue.
  - c.- Al menú principal del software
  - d.- Ninguna de las anteriores.



## NOTAS


## **SEGUNDA PARTE**

### **ELABORANDO EL MODELO DEL SISTEMA**

### **ACTIVIDADES**

#### **ASESORIAS**



Las asesorías son los encuentros para aclarar dudas, todo con objeto de concretar el informe y la construcción del software. El profesor asesor las planifica, ya que pueden ser por grupo o para toda la sección de UC-PST.

#### **LECTURAS**



En el contexto de este componente, **el estudiante de PST** debe apropiarse del contexto de REQUERIMIENTOS, MERINDE y UML, por lo tanto, debe realizar lecturas constantes a la medida que avanza el proyecto.

#### **INFOGRAFÍA**



El estudiante, una vez concluida la **FASE II – REQUERIMIENTOS, DISEÑO Y DESARROLLO DEL SOFTWARE**, debe preparar la infografía de la fase, como material de apoyo para el punto de control.

#### **AVANCE INFORME**



El **equipo de PST**, debe tomar el esquema que corresponde a su trayecto, y desarrollar paso a paso la **FASE II – REQUERIMIENTOS, DISEÑO Y DESARROLLO DEL SOFTWARE**, la cual debe entregar como indica el profesor asesor.

#### **DEMOSTRACIÓN PROTOTIPO.**



El **equipo de PST**, debe tomar el prototipo del software y presentar a la comunidad para **validar la adecuación de los requerimientos**.

# **SEGUNDA PARTE**

## **ELABORANDO EL MODELO DEL SISTEMA**

### **EVALUACIÓN**

#### **PUNTO DE CONTROL:**



**Punto de Control:** es un **encuentro planificado** por el profesor asesor, para **socializar el avance del informe** del PST y con ello **evidenciar la comprensión de los requerimientos y arquitectura del Software**. El equipo **debe apoyarse en un avance del Software** para su exposición, como material de apoyo que oriente el discurso de todos los integrantes. Participa el equipo de trabajo/docente asesor/jurado; esta actividad de vital importancia permite concretar algunos faltantes y conciliar las ideas sobre la propuesta. Se debe realizar en las instalaciones de la universidad. **Nota: el primer avance del software debe partir del HOMEPAGE y probar los formularios maestros y algunas transacciones.**

#### **AVANCE INFORME**



**El avance:** se centra en presentar en un informe: I - **REQUERIMIENTOS DEL SOFTWARE**; este punto debe contener 1.- **Especificación Requerimientos Funcionales** lo que incluye: Lista de Requerimientos/ Especificación de los requerimientos/ Diagrama de uso de uso del Sistema/plantillas de especificación de los casos de uso. 2.- **Especificación de requerimientos No funcionales**: aquí debe indicar lista y especificación. II - **DESARROLLO DE SOFTWARE**; este punto debe contener 1.- **Descripción del Software** Nombre /Objetivos /Alcance /Metodología utilizada. 2.- **Arquitectura del Software**, la cual se expresa mediante **Modelo de Diseño** (Diagrama de Objeto, Diagrama de Clase) /

**Modelo de Datos** (Modelo E/R y Diccionario de Datos)/ **Mapa de Navegación**/ Carta estructurada/ **Interfaz de usuario** (Formularios) / **Diagrama de componentes** y **Diagrama de despliegue**. NOTA: se recomienda que antes de iniciar la arquitectura, chequear bien los requerimientos.



**REALIZA EL AVANCE DEL INFORME:** es importante la participación de todos los integrantes del equipo, deben tener la comprensión del todo.



Para la **FASE II – REQUERIMIENTOS, DISEÑO Y DESARROLLO DEL SOFTWARE**, apoya aquí:

**I- REQUERIMIENTOS DEL SOFTWARE** << Consiste en **señalar los requerimientos que enmarcan el software**. En este orden de ideas, la definición de las **funcionalidades y características** del producto >>

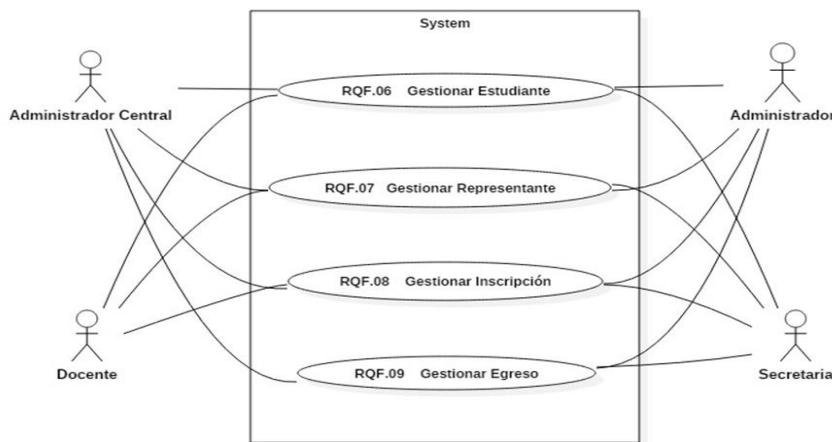
**Lista de Requerimientos.** << Consiste en presentar en el **cuadro para tal fin, la lista de requerimientos, tanto los señalados por el usuario, como los resultantes del análisis del contexto**. Ejemplo >>

Código	Nombre	Actores participantes
RQF-03	Gestionar Año Escolar	Administrador central, administrador.
RQF-05	Gestionar Personal	Administrador central, administrador, secretaria.
RQF-06	Gestionar Estudiante	Administrador central, administrador, secretaria, docente.

**Especificación de Requerimientos Funcionales:** << Consiste en presentar para cada requerimiento su descripción detallada como señala el cuadro para tal fin. Aquí un ejemplo>>

ID del Requerimiento:	RQF-03			
Nombre del Requerimiento:	Gestionar año escolar.			
Características:	Permite registrar, modificar y consultar un año escolar en el sistema.			
Descripción del requerimiento:	Solo los usuarios nivel administrador o el administrador central pueden gestionar el año escolar, es necesario que exista un año escolar activo para poder realizar diferentes operaciones en el sistema como: Inscripción, Egreso, Carga de Nota, Lapsos, etc.			
Prioridad del requerimiento:				
<input checked="" type="checkbox"/> Alta	<input type="checkbox"/> Media Alta	<input type="checkbox"/> Media	<input type="checkbox"/> Media Baja	<input type="checkbox"/> Baja

**Diagrama de Caso Uso del Software.** << Consiste en presentar en el diagrama de caso uso del software, todos los requerimientos funcionales nombrado en la listado. Aquí un ejemplo >>



**Especificaciones de los Casos de uso del Software** << Consiste en presentar para cada caso de uso su plantilla de especificación, debe detallarla muy bien>>.

**Especificación de Requerimientos No Funcionales** << Consiste en presentar la lista de requerimientos no funcionales con sus especificaciones. Aquí un ejemplo>>

Tipo de requisito	Código	Descripción
Restricciones de diseño	RQNF-01	El software web se desarrollará e implementará bajo estándares de calidad y pertinencia social, priorizando el uso de plataformas libres (software libre), utilizando el lenguaje de programación PHP y utilizará el estándar HTML para el diseño de las páginas web de sistema.
	RQNF-02	El motor de base datos se desarrollará bajo MYSQL.
	RQNF-03	El sistema debe utilizar los servicios de la red interna de la institución para establecer comunicación entre los clientes, el servidor web y manejador de base de datos.
Interfaces de usuario	RQNF-04	La aplicación debe mantener los estilos (colores, tipos de letra, entre otros) de la Institución.
	RQNF-05	La aplicación debe proporcionar al usuario opciones que permitan realizar el trabajo, y deben estar correctamente descritas.
Seguridad	RQNF-06	Acceso restringido a la aplicación mediante autenticación por usuario y contraseña. El sistema tendrá que evitar que el personal no autorizado, o sin permisos, pueda acceder a los datos privados de los usuarios, así como el acceso al sistema.
Usabilidad	RQNF-07	La aplicación debe proporcionar opciones claras para los usuarios. El manejo de la aplicación tiene que resultar lo más intuitivo posible, es decir la navegación por la aplicación tiene que ser clara para facilitar la operatividad al usuario.
Mantenimiento	RQNF-08	El sistema debe estar en capacidad de permitir en el futuro su fácil mantenimiento con respecto a los posibles errores que se puedan presentar durante la operación del mismo.
	RQNF-09	Todo el sistema deberá estar completamente documentado, tanto en el código fuente como en los manuales de administración y de usuario.
Compatibilidad	RQNF-11	El sistema deberá funcionar en cualquier ordenador y un explorador de gran difusión en el mercado, debe visualizarse correctamente por lo menos en los navegadores más extendidos.
Fuente: Integrantes del PST		

**II DESARROLLO DEL SOFTWARE << Consiste en señalar el Nombre del software /Objetivos / Alcance / Metodología de Desarrollo y la Arquitectura >>**

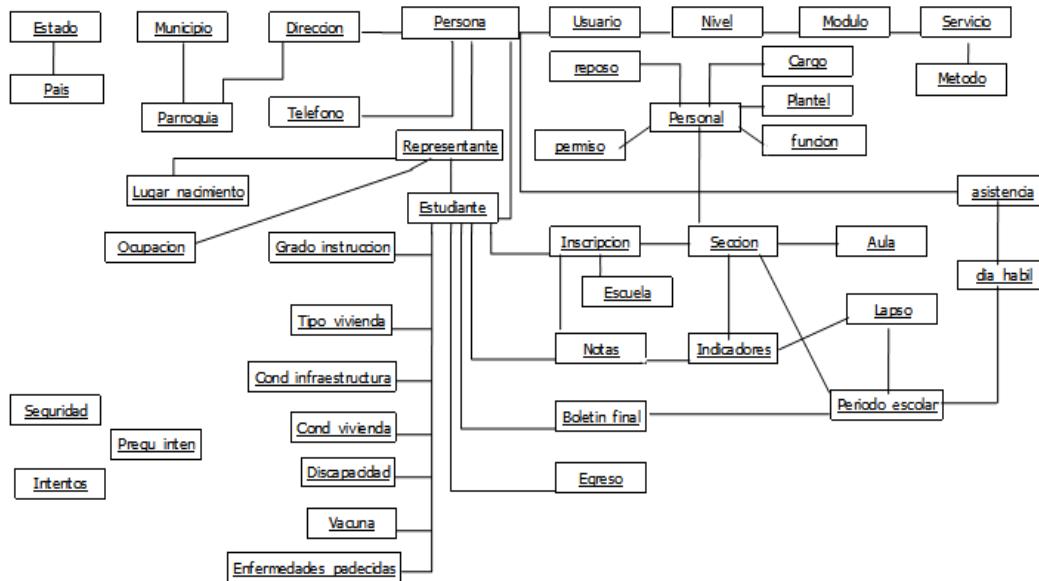
Nombre del Software: **Software para .....**

Objetivo General: **Aplicar un software para....., que**

Alcance: **permitirá realizar las operaciones relacionadas con la inscripción, rendimiento escolar y egreso; en el mismo orden un control de seguridad para el acceso al software**

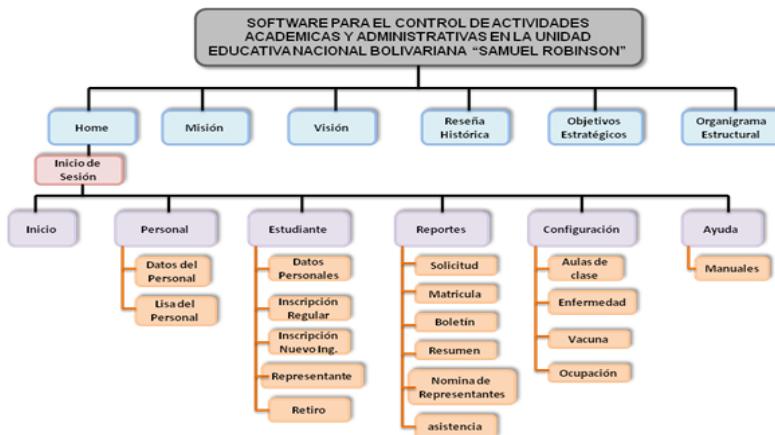
**Arquitectura del software** << Consiste en **señalar** el Modelo de Diseño/ Modelo de Datos/ Mapa de Navegación / Interfaz de Usuario y Diagramas de Componente y de Despliegue, aquí un ejemplo >>

**Modelo de Diseño**<< Consiste en **señalar** el Diagrama de Objeto y Diagrama de Clase >>



**Modelo de Dato** << Consiste en **señalar** el Modelo Conceptual y Físico de la Base de Dato, aunado al diccionario de datos **como se indicó en la sección de contenido**>>

**Mapa de Navegación** << Consiste en **señalar** la estructura del software desde el punto de vista de las funcionalidades, aquí un ejemplo >>



**Interfaz de usuario << Consiste en señalar los formularios que constituyen la comunicación usuario con el computados, aquí un ejemplo >>**



**Diagrama de componentes y Diagrama de despliegue << Consiste en presentar los diagramas que ilustren la relación de los componentes de software y componentes de hardware como plataforma para el funcionamiento del producto >>**

## TERCER COMPONENTE

### CONTENIDO

## TERCERA PARTE IMPLANTANDO EL SOFTWARE



**Un software de calidad debe cumplir en ser:**

**Correcto** - acorde a las especificación, **Confiable** - lo esperado por el usuario,

**Robusto** - tolerante a fallas. **Reusable** - pueda integrarse a otro software.

**Portable** - utilizable en diferentes plataformas, **Eficiente** – Uso adecuado de los recursos computacionales, **Comprensibles** - documentación fácil de entender el sistema. **Usable** - satisfacción del usuario para usarlo.

**Mantenible** - fácil de modificar y **Seguro** –proteger acceso a los datos.

## IMPLANTANDO EL SOFTWARE

### OBJETIVO(S)

- **Evaluar el Software**, para concretar la calidad del su desarrollo.



- **Implantar del Software**, para determinar la aceptación del usuario.
- Realizar la **Documentación del software**, para apoyar al usuario en su navegabilidad y al programadores en el mantenimiento del software

### TEMAS



- Pruebas del Software.
- Implementación del Software
- Factores de Riesgo.
- Documentación del Software.

### ACTIVIDADES



- Asesoría.
- Lectura: Pruebas e Implementación de Software.
- Infografía - Fases del Proyecto

### EVALUACIÓN

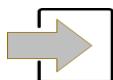


- Punto de Control Software
- Avance de Informe
- Manual de Usuario / Manual de Sistema

## TERCERA PARTE

### IMPLANTANDO EL SOFTWARE

#### TEMAS



**PRUEBAS DE SOFTWARE:** una vez finalizada la etapa de implementación (codificación del software), se abordan acciones para comprobar que los resultados esperados, son iguales a los resultados obtenidos. Todo, en función de las especificaciones dadas por el usuario. Por lo tanto, es fundamental aplicar pruebas sistemáticamente con objeto de alcanzar un software de calidad. En **MERINDE** se identifica **pruebas** como una disciplina que se realiza **durante la fase de construcción** y previo a implantación (Ver imagen 20).

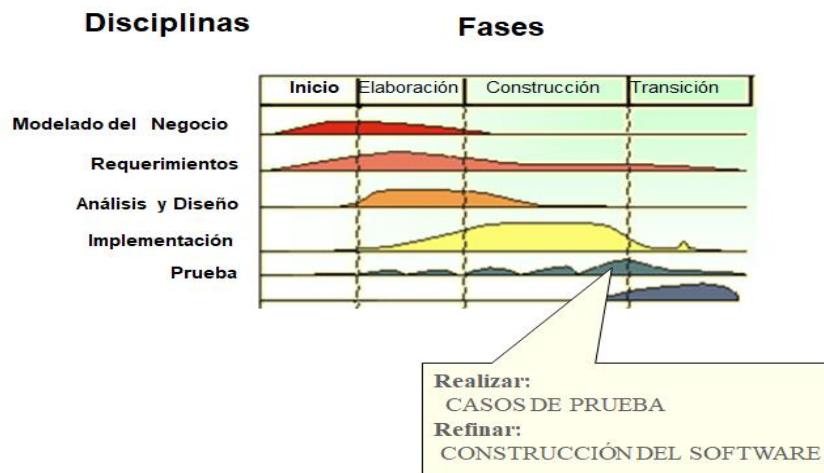


Imagen 20: Disciplina Pruebas. Fuente: MERINDE.



**Identifica importancia de las prueba de software.** Con tus compañeros comparte tu postura.



Las **PRUEBAS DE SOFTWARE** se abordan siguiendo los niveles de prueba como sigue:

1.- La **prueba de unidad** se centra en cada módulo individualmente, asegurando que funciona adecuadamente como un componente, hace uso de las técnicas de **prueba de caja blanca**, ejercitando caminos específicos de la estructura de control del componente.

2. La **prueba de Integración**, se ensamblan o integran los componentes para formar el paquete de software completo, se dirige a todos los aspectos asociados con el doble problema de verificación y de construcción del programa. Durante la integración, las técnicas que más **prevalecen son las de diseño de casos de prueba de caja negra**, aunque se pueden llevar a cabo algunas pruebas de caja blanca con el fin de asegurar que se cubren los principales caminos de control.

3.-La **prueba de validación** permite determinar si **satisface todos los requisitos funcionales**, de comportamiento y de rendimiento, aquí se usan exclusivamente técnicas de **prueba de caja negra**.

4. **La prueba del sistema** verifica que cada elemento encaja de forma adecuada y si alcanza la funcionalidad aunado el rendimiento del sistema total (ver **Imagen 21** de estrategia de pruebas).

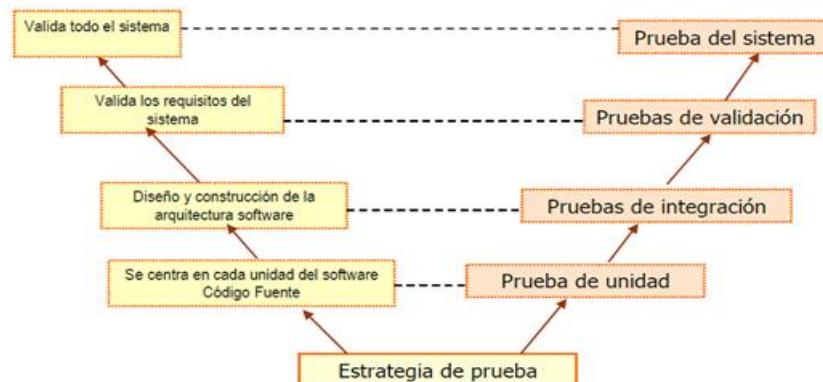


Imagen 21: Estrategia de Pruebas. Fuente Plessman (2006).



**Una prueba** debe centrarse en **dos objetivos**: 1) Probar si el software NO hace lo que debe hacer y 2) Probar si el software hace lo que no debe hacer. Una **prueba** debe descubrir el más alto nivel de errores.



**Analiza cada formulario del software.** En reunión con tu grupo de proyecto, identifica para cada formulario reglas del negocio, precondiciones, post condiciones, condiciones para gestión de base de dato. Todo ello es fuente de pruebas en el cumplimiento de los requerimientos

#### TAREAS PARA PROBAR UN SOFTWARE:

**1.-Diseñar los casos de prueba,** determinan un conjunto de entradas, condiciones de ejecución y resultados esperados para un objetivo particular.

**2.- Definir el procedimiento:** implica la especificación de cómo se va a llevar a cabo el proceso, quién lo va a realizar, cuándo, entre otros detalles.

**3. Ejecutar la prueba:** implica aplicar los casos de prueba identificando los posibles fallos o errores producidos al comparar los resultados esperados con los resultados obtenidos.

**4. Realizar informe de las pruebas,** con el resultado de la ejecución.

#### TIPOS DE PRUEBAS:

**1.-Pruebas de Caja Blanca:** método donde los Casos de Prueba atiende el comportamiento interno y la estructura del programa, **examina la lógica interna del código de programación).**

**2.-Pruebas de Caja Negra o Funcionales:** método que determina el comportamiento del software, solo estudiando sus entradas y las salidas,

para identificar entradas que causen un comportamiento erróneo. (Ver **Imagen 22**: Técnicas de Diseño de Casos de Prueba.



En el contexto de **PST**, se recomienda realizar **pruebas de unidad** para todos los **maestros**, en primera instancia de caja negra y si da error, aplicar pruebas caja blanca.

Las **pruebas de integración a todas las transacciones, reportes**, igualmente primero caja negra y luego caja blanca si lo amerita.

Las **pruebas de sistema**, esta se realiza una vez que se integra todo los componentes.

Finalmente, las **pruebas de aceptación** por el **usuario**, la cual verifica las funcionalidades con datos reales.

En este orden, el líder de proyecto debe **planificar las pruebas**, todos los componentes (formularios) deben ser reflejados en el cronograma de pruebas.

**La criticidad indica el estatus del componente que está en pruebas**, en **rojo** (componente por probar), **amarillo** (componente aun en pruebas), **verde** (componente consolidado – información confiable). En **cuadro 17** se ilustra la plantilla de planificación de pruebas y **cuadro 18** cronograma de pruebas para sistematizar el proceso.



**Planifica las pruebas.** En reunión con los integrantes del equipo y en algunos casos con representantes de la comunidad, con objeto de establecer las jornadas de prueba.

**Cuadro 17: Plantilla para la planificación de pruebas.**

TIPO DE PRUEBA	OBJETIVO	PARTICIPANTES	FECHA DE EJECUCIÓN	AMBIENTE	MÉTODO
<b>UNITARIAS</b>	Detectar errores en la especificación y lógica de los Algoritmos.			Desarrollo	Caja blanca
<b>INTEGRALES</b>	Detectar errores en la correcta unión de los componentes entre sí a través de sus interfaces y si cumple con la funcionalidad establecida			Desarrollo	Caja blanca
<b>FUNCIONALES</b>	Detectar errores en la implementación de requerimientos			Desarrollo	Funcional
<b>DE SISTEMA</b>	Detectar fallas en el cumplimiento de los requerimientos como lo hará el usuario final			Desarrollo	Funcional
<b>ACEPTACIÓN</b>	Detectar fallas cuando el usuario compruebe que el sistema hace lo especificado en el contrato para dar el visto bueno.			Producción	Caja negra/ Funcional

Fuente: Canela, M (2023).

**Cuadro 18: Plantilla cronograma de pruebas.**

CRITICIDAD	ACTIVIDADES	RECURSOS	MES -1	MES-2		MES 3	
<b>NIVEL SEGURIDAD</b>							
● ○ ○	Prueba 1						
	Prueba 2						
<b>MAESTROS</b>							
○ ○ ○	Prueba 3						
<b>TRANSACCIONALES</b>							
○ ○ ○	Prueba						
	Prueba						
<b>REPORTES</b>							
○ □ ■	Prueba						
	Prueba						
<b>MANTENIMIENTO</b>							
○ □ ■	Prueba						
<b>ACEPTACIÓN</b>							
○ △ ■	Prueba						

Fuente: Canela, M (2023).



Ahora bien, **en el contexto de PST**, para cada prueba se **debe diseñar un caso de prueba (CP)**; en él se señalan las acciones para verificar una característica o una funcionalidad del software. En la plantilla debe indicar: código RF/nombre/propósito/fecha /hora/ validador/ datos

entradas/ resultado esperado/ resultado obtenido/ Evaluación (criticidad). Ver **cuadro 19**, ilustra plantilla Caso de Prueba.

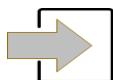
#### Cuadro 19: Plantilla Caso de Prueba

CASO DE PRUEBA	
Código RF	
Nombre RF	
Propósito de la prueba:	
Fecha / Hora de la Prueba:	
Validador (es):	
DATOS ENTRADA DE LA PRUEBA	
RESULTADOS ESPERADOS	RESULTADOS OBTENIDOS
ERRORES ASOCIADOS	
CORRECCIÓN DE ERRORES	
EVALUACIÓN DE LA PRUEBA	

Fuente: Canela, M (2023).



Diseña los casos de pruebas y emprende las pruebas con la participación de cada integrante del equipo, cada representante de la comunidad, e invitados. Tomar datos reales y supuestos.



**IMPLANTACIÓN DEL SOFTWARE:** proceso en el cual se procede a la inserción del software en la plataforma tecnológica de la organización, con objeto que el usuario se apropie de él operándolo. Para ello, se realiza la identificación de los equipos, configuración del hardware, instalación del producto, capacitación del usuario, conversión de datos, pruebas de implantación y entrega de la documentación.



En PST, se recomienda elaborar una planificación de **implantación**, donde se detallen las actividades, su objetivo, responsables, fecha inicio, fecha final. A continuación **cuadro 20**, ejemplo de plantillas para la planificación de la implantación.

#### Cuadro 20: Plantilla para la PLANIFICACIÓN DE LA IMPLANTACIÓN:

CONFIGURACIÓN DEL HARDWARE	
Objetivo	
Actividades	
Recursos	
Responsable(s)	
Fecha de inicio	
Fecha final	
INSTALACIÓN DEL SOFTWARE	
Objetivo	
Actividades	
Recursos	
Responsable(s)	
Fecha de inicio	
Fecha final	
CAPACITACIÓN DE USUARIO	
Objetivo	
Actividades	
Recursos	
Responsable(s)	
Fecha de inicio	
Fecha final	
CONVERSIÓN DE DATOS	
Objetivo	
Actividades	
Recursos	
Responsable(s)	
Fecha de inicio	
Fecha final	



**Planifica y emprende la implantación**, esta fase se realiza en la comunidad, en la plataforma tecnológica asignada para ello. Es posible encontrar aun errores, resuelve entes de la socialización. **Toma evidencias** de todo encuentro con la comunidad.

**RESULTADOS DE LA IMPLANTACION DEL PROYECTO:** en este contexto, para asegurar la calidad del software, es importante aplicar durante el proceso de implantación **las métricas de evaluación ISO/EC 9126** para garantizar el correcto funcionamiento de las operaciones en consonancia con los requerimientos de la comunidad. A continuación en **cuadro 21**, se ilustra plantilla para evaluar las métricas.

**Cuadro 21**, Plantilla de métricas.

MÉTRICAS EXTERNAS DE CALIDAD.				
USABILIDAD			RESPUESTAS	
ELEMENTO	MÉTRICA	PROPONE MEDIR	SI	NO
Aprendizaje	Familiarización con la estructura del sistema			
	Tiempo de aprendizaje			
Atractividad	Aceptación de los colores en el software			
	Aceptación de la interfaz			
Comprendibilidad	Comprensibilidad del menú			
	Amigabilidad con el lenguaje			
	Eficiencia de los textos de ayuda			
Operabilidad	Adaptación a las paginaciones			
	Accesibilidad a las operaciones			
FUNCIONABILIDAD				
ELEMENTO	METRICA.	PROPONE MEDIR	SI	NO
Adecuación	Fiabilidad de resultados			
	Disposición de funciones			
Exactitud	Precisión de resultados			
	Confiabilidad de resultados			
Seguridad	Seguridad de usuario			
	Seguridad de contraseña			

<b>Cumplimiento Funcional</b>	Estándares de funcionalidad			
	Regulaciones de funcionalidad			
<b>FIABILIDAD</b>				<b>RESPUESTA</b>
<b>ELEMENTO</b>	<b>MÉTRICA</b>	<b>PROPONE MEDIR</b>	<b>SI</b>	<b>NO</b>
<b>Recuperabilidad</b>	Desempeño			
	Capacidad de resguardo de datos Capacidad de mantenerse estable y operativo a pesar de fallas			
<b>Tolerancia A Fallos</b>	Comportamiento ante fallos permanentes			
	Grado de solución ante fallos totales			
<b>Cumplimiento de Fiabilidad</b>	Madurez de las pruebas			
	adaptabilidad a normas de fiabilidad			
<b>EFICIENCIA</b>				
<b>Comportamiento en el tiempo</b>	Tiempo de respuesta			
	Cumplimiento de la eficiencia			
<b>Comportamiento de recursos</b>	Tiempo de uso en los recursos			
	Utilización de recursos			
<b>CALIDAD EN USO</b>				
<b>Eficacia</b>	Objetivos del software			
<b>Productividad</b>	Eficaz y eficiente			
	Eficaz, pero no eficiente			
<b>Seguridad</b>	Seguridad de datos			
	Seguridad del sistema			
<b>Satisfacción</b>	Escala de satisfacción			



En este contexto de evaluar el software implantado **se sugiere en PST**, una vez aplicado el instrumento a los usuarios, realizar un análisis de datos de forma porcentual de las respuestas obtenidas



**Emprende la evaluación del software** desde las métricas de calidad, presenta los resultados.



Otro elemento que especificar es la **ESTRUCTURA LÓGICA Y FÍSICA DE LA RED**: aquí se debe mostrar los mapas físicos y lógicos de la arquitectura de la red donde debe estar alojado el software desarrollado.

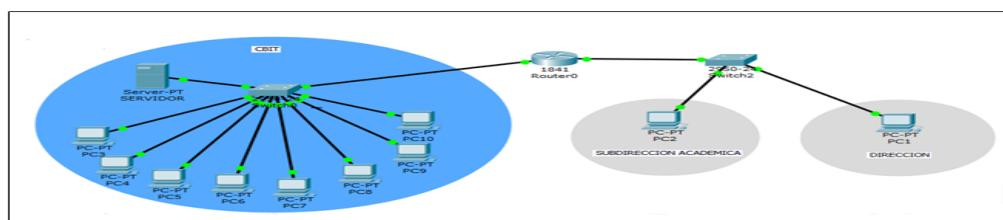
**Diagrama Lógico de la Red:** Se refiere a la conexión que existe entre equipos ya sea directa o indirectamente. Este diagrama es importante ya que permite establecer un orden de conexión de manera segura y eficaz.

**Diagrama físico de red** muestra la disposición física de los equipos con los que se cuenta a la hora de realizar la distribución y conexión de los mismos, ya sea de punto a punto o multipunto, de acuerdo a lo que sea requerido.

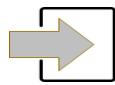


**Elabora el diagrama físico y lógico de la red.** Ella debe estar en función de los componentes tecnológicos donde funcionara el software.

#### **Diagrama Lógico de la Red:**



#### **Diagrama Físico de Red**

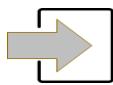


**FACTORES DE RIESGOS:** Un riesgo es un **evento o condición incierta que, si sucede, afecta algún objetivo del proyecto**. Se debe

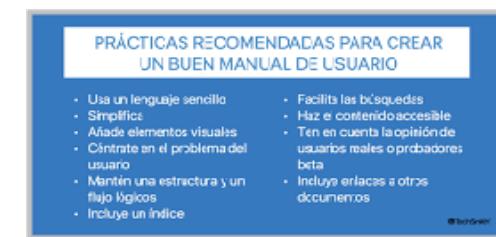
indicar según su caso de amenazas que afectara el desarrollo y funcionamiento del software.

### EJEMPLO:

IDENTIFICACIÓN DE RIESGOS	
TECNOLÓGICO	
ECONÓMICO	
FINANCIERO	
SOCIAL	
OPERACIONAL	
LEGAL	
TIEMPO	
AMBIENTAL	



**DOCUMENTACIÓN DEL SOFTWARE:** son elementos fundamentales que deben acompañar un software; **permiten guiar a las personas sobre** el funcionamiento de los aplicativos y de la solución a los problemas más comunes. Está compuesto por el **manual de usuario** (guía de navegación y de operación del software dirigido al usuario final), **manual de instalación** (guía práctica para la instalación del software), **manual de sistema** (guía que requiere un programador para darle mantenimiento; parte desde los requerimientos, arquitectura lógica y física del sistema, plan de pruebas realizadas).



**Elabora los manuales y documenta el código** de los componentes del software.



**EVALUACION:** Indica en el paréntesis la letra de la opción correcta:

**1.- Una prueba de unidad: (\_\_\_\_)**

- a.- Se aplica a los formularios de transacciones.
- b.- Se aplica a los formularios maestros.
- c.- Es aplicada en el último nivel de pruebas.
- d.- Ninguna de las anteriores.

**2.- Una prueba Caja Blanca: (\_\_\_\_)**

- a.- Permite auditar el código.
- b.- Permite identificar debilidades en el arte de la programación.
- c.- Es recomendable utilizarla en la etapa de implantación.
- d.- Ninguna de las anteriores

**3.- La capacitación del usuario se contempla en (\_\_\_\_)**

- a.- Planificación pruebas de desarrollo.
- b.- Planificación pruebas de Implantación
- c.- Planificación de desarrollo de software
- d.- Ninguna de las anteriores.

**4.- Un caso de Prueba es (\_\_\_\_)**

- a.- Es un artefacto de UML
- b.- Es un artefacto de MERINDE
- c.- Es una plantilla para organizar una prueba
- d.- Ninguna de las anteriores.

**5.- El manual de usuario (\_\_\_\_)**

- a.- Oriente al programador para dar mantenimiento al software
- b.- Orienta el uso del software.
- c.- Orienta como instalar el software
- d.- Ninguna de las anteriores.

**6.- Las Métricas de Calidad (\_\_\_\_)**

- a.- Permite verificar la adecuación de los requerimientos.
- b.- Permite verificar la aceptación de los usuarios del producto.
- c.- Permite identificar fallas en la programación del software.
- d.- Todas las anteriores.

**7.- Los factores de riesgos (\_\_\_\_)**

- a.- Identifican dificultades para el buen funcionamiento del software.

- b.- Permiten dar recomendaciones ante dificultades de funcionamiento.
- c.- Permiten reestructurar plataforma del sistema.
- d.- Ninguna de las anteriores.

**8.- La implantación del software (\_\_\_\_)**

- a. Implica tomar los requerimientos y crear la arquitectura del software.
- b.- Se pone en práctica el arte de la programación de software.
- c.- Implica la capacitación de los usuarios.
- d.- Ninguna de las anteriores.

	<b>NOTAS</b>

## **TERCERA PARTE**

### **IMPLANTANDO EL SOFTWARE**

#### **ACTIVIDADES**

##### **ASESORIAS**



Las asesorías son encuentros para aclarar dudas respecto a concretar el avance proyecto. El profesor asesor las planifica, ya que pueden ser por grupo o para toda la sección de la UC-PST.

##### **LECTURAS**



En el contexto, **el estudiante de PST** debe apropiarse del contexto de PRUEBAS DEL SOFTWARE (durante el desarrollo y durante la implantación), MERINDE y UML, por lo tanto, debe realizar lecturas constantes a la medida que avanza el proyecto.

##### **INFOGRAFÍA**



El estudiante, una vez concluida la **FASE I, FASE II, FASE III y FASE IV** debe preparar la infografía, como material de apoyo para el punto de control.

##### **PUNTO DE CONTROL**



**Punto de Control:** es un **encuentro planificado** por el profesor asesor, para **socializar el avance del informe** del PST y con ello **evidenciar la comprensión del 100% Software**. El equipo **debe apoyarse en el Software** para su exposición, como material de apoyo que oriente el discurso de todos los integrantes. Participa el equipo de trabajo/docente asesor/jurado; esta actividad de vital importancia permite concretar algunos faltantes y conciliar las ideas sobre la propuesta. Se debe realizar en las

instalaciones de la universidad. **Nota:** El software debe partir del **Homepage** y probar **TODAS** las funcionalidades, aunado a creación de usuarios.

### **AVANCE DEL INFORME**



El avance del informe, se centra en la **FASE III - IMPLANTACIÓN, PRUEBAS Y OPTIMIZACIÓN DEL PROYECTO**. Esta debe presentar en primer lugar como se organizaron y se aplicaron las pruebas al software, para ello debe indicar su planificación, casos de prueba y análisis de resultados; en este orden, el plan de implantación del software y logros obtenidos: **NOTA:** se recomienda que antes de iniciar la implantación, chequear bien todas las funcionales.



**REALIZA EL AVANCE DEL INFORME:** es importante la participación de todos los integrantes del equipo, deben tener la comprensión del todo.



Para el avance de la **FASE III - IMPLANTACIÓN, PRUEBAS Y OPTIMIZACIÓN DEL PROYECTO**, aquí orientaciones

### **FASE III - IMPLANTACIÓN, PRUEBAS Y OPTIMIZACIÓN DEL PROYECTO**

**3.1.- DETECCION Y CORRECCION DE ERRORES DEL PROYECTO. <<**  
Consiste en **señalar** la Planificación de las Prueba del Software, Diseño de Casos de pruebas e indicar el análisis de los Resultado de pruebas del software >>

**Planificación de prueba del software:**<< aquí debe insertar la Planificación de Pruebas del software a nivel de unidad, integración y de sistema, y aceptación. La misma debe estar indicada en la Plantilla para la planificación de pruebas. Igualmente expresar todos los componentes del software expresados en la plantilla Plantilla cronograma de pruebas., es decir todos los formularios como medio para activar las funcionalidades del software >>

TIPO DE PRUEBA	OBJETIVO	RESPONSABLES	FECHA DE EJECUCIÓN.	AMBIENTE	MÉTODO
Unidad	Detectar errores en la especificación y lógica de los algoritmos.	Programadores (xxxxxxxx)		Desarrollo	Caja blanca
Integral	Detectar errores en la correcta unión de los componentes entre sí a través de sus interfaces y si cumple con la funcionalidad establecida.	Programadores (xxxxxxxx)		Desarrollo	Caja blanca
Funcionales	Detectar errores en la implementación de requerimientos	Analistas de Pruebas (xxxxxxxx)		Desarrollo	Funcional
De Sistema	Detectar fallas en el cumplimiento de los requerimientos como lo hará el usuario final	Analistas de Pruebas (xxxxxxxx)		Desarrollo	Funcional
Aceptación	Detectar fallas cuando el usuario compruebe que el sistema hace lo especificado en el contrato para dar el visto bueno	Analistas de pruebas, Cliente (xxxxxxxx)		Producción	Funcional

Actividades	Recursos	Noviembre	Diciembre	Enero	Febrero				Marzo		
					\$1 Del 02 al 08	\$2 Del 09 al 15	\$3 Del 16 al 22	\$4 Del 23 al 28	\$1 Del 02 al 08	\$2 Del 09 al 15	\$3 Del 16 al 22
Pruebas para detección de errores en la lógica de algoritmos	Programador				Proceso continuo						
Pruebas integrales para detección de errores en interfaces y componentes	Programador				Proceso continuo						
<b>Seguridad</b>											
Realizar pruebas de creación de usuarios	Analista										
Realizar pruebas de parametrización de roles y perfiles	Analista										
Realizar pruebas de asignación de roles y perfiles a usuarios	Analista										
Realizar pruebas de inicio de sesión	Analista										
Realizar pruebas de accesos de usuarios	Analista										
Realizar pruebas de cambio / caducidad de contraseñas	Analista										
Realizar pruebas de restauración de datos de acceso	Analista										
Realizar pruebas de módulo de auditoría de operaciones	Analista										
<b>Datos maestros</b>											
Realizar pruebas de funcionalidad de Maestra Personal	Analista										
Realizar pruebas de funcionalidad de Maestra Materias	Analista										
Realizar pruebas de funcionalidad de Maestra Aulas	Analista										
Realizar pruebas de funcionalidad de Maestra Profesor	Analista										
<b>Transacciones</b>											
Realizar pruebas de apertura de proceso inscripción	Analista										
Realizar pruebas de conformación de horarios	Analista										
Realizar pruebas de adquisición y asignación de Bienes	Analista										
Realizar pruebas de préstamo y devolución de libros	Analista										
Realizar pruebas de asignación de estudiantes a secciones	Analista										
Realizar pruebas de emisión de horarios por profesor	Analista										
Realizar pruebas de colisiones de disponibilidad	Analista										
<b>Reportes</b>											
Realizar pruebas de reportes de datos maestros	Analista										
Realizar pruebas de formulario de Comprobante de inscripción	Analista										
Realizar pruebas de Formularios de Bienes y Biblioteca	Analista										
Realizar pruebas de listado consulta bitácora para auditoría	Analista										
<b>Mantenimiento</b>											
Validar los mensajes del sistema	Analista										
Validar la integridad de la data con Respaldo / Restauración	Analista										
<b>Aceptación</b>											
Demo y pruebas finales con organización	Analista / Cliente										

**Diseño de Casos de pruebas:** << aquí debe insertar los casos de prueba más significativos, es decir donde encontró fallos o errores >>

IDENTIFICADOR :	RQF-01	RF-04	RF-04	RF-04	RF-04
Requerimiento Funcional:					
Propósito de la prueba:	Identificador: Requerimiento Funcional:				
Fecha / Hora de la Prueba:	Propósito de la prueba:	Identificador: Requerimiento Funcional:	Servicio Personal	Identificador: Requerimiento Funcional:	Servicio Personal
Validador (es):		Propósito de la prueba:	Validar denominación correcta que se registró en el sistema.	Propósito de la prueba:	Validar que el formulario Maestro denominado Personal funcione correctamente, de acuerdo a la información que se requiere capturar para el proceso de registro de las personas que harán uso del sistema.
Entrada:	Datos de Entrada:	Fecha / Hora de la Prueba:	06/10/2014 / 07:56 pm	Fecha / Hora de la Prueba:	06/10/2014 / 07:56 pm
	R: Validador (es):	En:	Analista	Validador (es):	Analista de pruebas
	xxxxxx				
Error:	Datos de Entrada para Pulsamos el Modulo Administrador despliega el formulario que nos permite operaciones a los datos relacionados con el personal registrado o las personas registradas en el sistema.	Entrada:			
Corrección:	Pulsamos el Modulo Administrador despliega el formulario que nos permite operaciones a los datos relacionados con el personal registrado o las personas registradas en el sistema.				
Evaluado:	Resultado esperado y obtenido: En la opción listar registros nos arrojaba el siguiente error.	Resul:			
	Error 1 Error en la consulta de datos de personas p INNER JOIN tpa ON p.cod_pais = tpa.cod_pais INNER JOIN tpersonalespr ON p.cod_pais = tpersonalespr.cod_pais ON pr.cod_rol = r.cod_rol "%ESTUDIANTE%" ORDER BY cedula	Errores:			
	Error en la consulta de conteo de personas p INNER JOIN tpa ON p.cod_pais = tpa.cod_pais INNER JOIN tpersonalespr ON p.cod_pais = tpersonalespr.cod_pais ON pr.cod_rol = r.cod_rol WHERE "%ESTUDIANTE%" ORDER BY cedula DESC. MySQLdijic Column "cedula" in order clause is ambiguous	Corrección:			
	No mostraba permiso Visualizar	Evaluació:			
		No mostraba permiso Visualizar	Corrección de Errores		
			Evaluación de la Prueba		
			No mostraba permiso Visualizar los registros en una lista.		

. **Resultado de pruebas del software**<< aquí debe insertar el análisis de resultados referidos a las pruebas de unidad, de integración, de sistema y aceptación >>

**3.2.- ESTRATEGIAS DE IMPLANTACION DEL PROYECTO.** << Consiste en **señalar** la Planificación de la implantación, Pruebas de Implementación, Resultados de implantación (Métricas) >>

**Planificación de la Implantación:** << aquí debe insertar la planificación de la implantación y resultados a cada actividad >>

**Pruebas de implantación:** << aquí debe insertar la planificación de las pruebas de implantación y resultados a cada actividad >>

**Resultados de la Implantación (Métricas)** << aquí debe insertar el análisis de resultados en la aplicación de las métricas

**3.3.- RESULTADOS Y LOGROS DEL PROYECTO** << Consiste en **realizar una descripción general del producto**, insertar el mapa físico y lógico de la red, factores de riesgo y los manuales >>

## CUARTO COMPONENTE

### EVALUACIÓN DEL CURSO

## CUARTO COMPONENTE

### EVALUACIÓN DEL CURSO



El seguimiento y control del proyecto, permiten identificar los avances concretos, apropiación del conocimiento y la adecuación de errores detectados.

## EVALUACIÓN DEL CURSO

### OBJETIVO(S)



- Identificar la **planificación académica** y de evaluación, para conocer las ponderaciones y tipo de evaluación durante el curso.
- Organizar las **plantillas de seguimiento del PST**, para tener evidencias de las evaluaciones al cierre por semestre y por trayecto.

### Temas



- 4.1.- Planificación Académica y de Evaluación.
- 4.2. -Ponderación de Evaluaciones.
- 4.3.-Seguimiento del Proyecto

### ACTIVIDADES



- Punto de Control
- Avance del Informe
- Pre-socialización
- Socialización

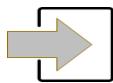
### EVALUACIÓN



- Seguimiento del Proyecto.
- Cierre del Trayecto.

## **CUARTO COMPONENTE - EVALUACIÓN EL CURSO**

### **TEMAS**



#### **PLANIFICACION ACADÉMICA Y DE EVALUACIÓN:**

documento que presenta la organización estructural del contenido del curso, los objetivos, estrategias pedagógicas, ponderación y tipo de evaluación; todo ello con objeto garantizar el éxito durante el proceso de enseñanza y aprendizaje.



En el **contexto de la UC-PST**, la cual se ubica en **un (1) Trayecto**; lo constituyen dos **(02) semestres** y cada uno debe tener una planificación académica y de evaluación. Cabe destacar que las actividades principales se centran en avances del informe con evaluación cualitativa y el informe final es cuantitativa, igualmente la Presocilizacion es cualitativa y la Socialización cuantitativa. Aquí se presenta la planificación orientadora por semestre para el trayecto.



**Analiza la planificación académica**, entrega las asignaciones en el tiempo previsto. Se evalúa responsabilidad, participación, dominio del contenido, dominio del contexto del proyecto y del software. Asiste a todas las actividades, ellas se toman en cuenta en la evaluación. Visita la comunidad y toma evidencias.

## PLANIFICACION SEMESTRE I DE LA UC-PST

SEMESTRE I			
CORTE	UNIDAD/ OBJETIVO / CONTENIDO	%	ACTIVIDAD
I	<b>ELABORACIÓN DE LA PROPUESTA</b> / Realizar el diagnóstico participativo, para identificar la situación que requiere una aplicación informática/ Levantamiento de Información, análisis de necesidades, Planificación y organización por áreas de saberes: Hardware y Software. /Planificar el proceso para administrar la aplicación en función de la calidad de la aplicación informática/ Factores de Riesgos.	5	Avance Informe. Punto de Control
II	<b>EJECUCIÓN DE PROYECTO</b> / Planificar las actividades, para abordar la solución de la problemática planteada/ Planificación de Proyecto, Cronograma de Actividades, Estudio de Factibilidad/ Aplicar la metodología de desarrollo de software seleccionada, para plantear el sistema/ / Metodología de Desarrollo de Software (MERINDE). Modelo del Negocio, Requerimientos, Arquitectura de Software. Modelo del Sistema. Construcción Software.	5	Avance Informe Punto de Control
III	<b>IMPLANTACIÓN – PRUEBAS DEL PROYECTO</b> / Aplicar técnicas de programación e implantación en la solución informática/ Estrategia de implantación del proyecto, Ejecución y evaluación de la implantación /Estrategia de implantación del proyecto, Planificación y aplicación de las pruebas, Detección y Corrección de errores.	5	Avance Informe Punto de Control <b>Pre socialización</b>
IV	<b>DEMOSTRACIÓN AVANCE del PST IV</b> / Socializar el avance aplicación informática, para evaluar su impacto social / Evaluación Informe, Evaluación Institucional del Software, Evaluación Comunidad del Software, Autoevaluación del estudiante en desempeño del PST.	10 10 10 5	Informe Final <b>Socialización</b> Autoevaluación

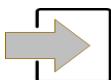
Fuente: Canela, M (2023).

## PLANIFICACION SEMESTRE II DE LA UC-PST

SEMESTRE II			
CORTE	UNIDAD/ OBJETIVO / CONTENIDO	%	ACTIVIDAD
I	<b>CONSTRUCCIÓN – PRUEBAS DEL PROYECTO</b> / Aplicar técnicas de programación en la solución informática, realizando pruebas para verificar que genera información confiable/ Metodología de Desarrollo de Software (MEINDE), Modelo del Sistema, Planificación y aplicación de las pruebas, Detección y Corrección de errores.	5	Avance Informe. Punto de Control
II	<b>CONSTRUCCIÓN – PRUEBAS DEL PROYECTO</b> / Aplicar técnicas de programación en la solución informática,	5	

	<i>realizando pruebas para verificar que genera información confiable/ Metodología de Desarrollo de Software, Modelo del Sistema, Planificación y aplicación de las pruebas, Detección y Corrección de errores/ demostración del avance del producto.</i>		Avance Informe Punto de Control
III	<b>IMPLANTACIÓN - MANUALES DEL PROYECTO/</b> <i>Aplicar lineamientos en la aplicación del proyecto, igualmente elaborar documento paso a paso que orienten a los usuarios abordar la aplicación informática, aunado a su forma de instalación /factores de riesgos/ presentación de los manuales de usuario y de sistema.</i>	5	Avance Informe Punto de Control Pre socialización
IV	<b>DEMOSTRACIÓN Y ENTREGA DEL PRODUCTO FINAL del PST II A LA COMUNIDAD/</b> <i>Socializar la aplicación informática, para evaluar su impacto social Evaluación Informe final, Evaluación jurado Institucional del Software, Evaluación Comunidad del Software, Autoevaluación del estudiante en desempeño del PST.</i>	10 10 10 5	Informe Final Socialización Autoevaluación

Fuente: Canela, M (2023).



**PONDERACIÓN DE EVALUACIÓN:** representa el porcentaje (%) asignado a una evaluación sobre un 100%. Toda planificación de evaluación distribuye e indica los valores.



**La ponderación de las evaluaciones al cierre de la UC-PST,** se encuentra normada para cada estudiante en un 100% como se ilustra en el **cuadro 20:** planificación de evaluaciones UC-PST- cierre trayecto, y se realiza para cada equipo de proyecto.

**Cuadro 20:** Planificación de Evaluación de la UC-PST – Cierre Trayecto.

Nº	Nombres	Apellidos	C.I.	SOCIALIZACION		Auto Eva	Puntos de Control	Informe Técnico Final	Nota Final/100%	Nota Final/20 PTS	Firma Estudiante
				COMUNIDAD	JURADO						
				20%	30%						
1											

Fuente: Coordinación PNFI (2018)



Es importante aclarar que la **ponderación de las evaluaciones para el cierre de cada semestre es de un 50%** y resultan del seguimiento. El cuadro 21, ilustra la planificación de evaluaciones UC-PST – cierre de semestre.

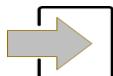
### El cuadro 36: planificación de evaluaciones UC-PST por semestre

Nº	Nombre s	Apellidos	C.I .	SOCIALIZACION		Auto Eva	Puntos de Control	Informe Técnico Final	Nota Final/50%	Firma Estudiante
				COMUNIDAD	JURADO					
				10%	15%					
1										

Fuente: Canela, M (2023).



**Analiza los porcentajes de las evaluaciones**, por semestre y por trayecto



**SEGUIMIENTO DEL PROYECTO:** consiste en chequear el cumplimiento de las actividades indicadas en la planificación de evaluaciones, las cuales permiten evidenciar el trabajo continuo del equipo de proyecto. Es monitoreada por el docente asesor o docente que suministra la UC-PST.



En el **contexto de la UC-PST**, se realizan sobre: avances del informe, avances del software, pre socialización y socialización del PTS. Se indican **Plantillas para el seguimiento del semestre I y Semestre II**.

## Plantilla seguimiento INFORME – semestre I

N	Apellido s	Nombre s	C.I.	INFORME							
				Avance I (Fase I - borrador )		Avance II (Fase I corregida, Fase II borrador)		Avance III Fase I, II corregida Fase III - Borrador		Informe Final Semestre I Fase I, II Concluidas Fase III – aun con faltante	
				Entregó	Estatus	Entregó	Estatus	Entregó	Estatus	Entregó	Estatus
				Si/NO	D/EP/C	Si/NO	D/EP/C	Si/NO	D/EP/C	S/N	D/EP/C
1											

Fuente: Canela, M (2023).



Identifica el seguimiento de las entregas referidas al informe durante cada semestre, su alcance.

## Plantilla seguimiento INFORME – semestre II

N	Apellido s	Nombre s	C.I.	INFORME							
				Avance I Fase I, II Concluidas Fase III – concluidas		Avance II (Fase IV borrador)		Avance III Manuales Borradores		Informe Final Semestre II Fase I, II, III , IV concluidas Manuales concluidos	
				Entregó	Estatus	Entregó	Estatus	Entregó	Estatus	Entregó	Estatus
				Si/NO	D/EP/C	Si/NO	D/EP/C	Si/NO	D/EP/C	S/N	D/EP/C
1											

Fuente: Canela, M (2023).



**Identifica el seguimiento referido al desarrollo del software durante cada semestre, su alcance.**

### **Plantilla seguimiento SOFTWARE – semestre I**

N	Apellido s	Nombre s	C.I.	SOFTWARE								
				Punto CONTROL I Avance Software		Punto CONTROL II Avance Software		PRE-SOCIALIZACION producto consistente a los requerimientos y atributos de calidad		SOCIALIZACIÓN producto consistente a los requerimientos y atributos de calidad en desarrollo		
				Asistió	Estatus	Asistió	Estatus	Asistió	Estatus	Asistió	Nota Jurado/ 15 puntos	Nota comunitad /10 puntos
1												
2												

Fuente: Canela, M (2023).

### **Plantilla seguimiento SOFTWARE – semestre II**

N	Apellido s	Nombre s	C.I.	SOFTWARE								
				Punto CONTROL I Avance Software		Punto CONTROL II Avance Software		PRE-SOCIALIZACION producto consistente a los requerimientos y atributos de calidad		SOCIALIZACIÓN producto consistente a los requerimientos y atributos de calidad - implantado		
				Asistió	Estatus	Asistió	Estatus	Asistió	Estatus	Asistió	Nota Jurado/ 15 puntos	Nota comunitad /10 puntos
1												
2												

Fuente: Canela, M (2023).



**Identifica el seguimiento referido a la asistencia y participación.**

### Plantilla seguimiento ASISTENCIA Y PARTICIPACIÓN – semestre I

N	Apellidos	Nombres	C.I.	ASISTENCIA - PARTICIPACIÓN									
				Asesoría 1		Asesoría 2		Ensayo		Infografía Fases I,II, y III		PRE SOCIALIZACION	
				Asistió	Estatus	Asistió	Estatus	Entregó	Estatus	Entrego SI /No	Estatus D/EP/C	Asistió SI /No	Estatus D/EP/C
				Si/NO	D/EP/C	Si/NO	D/EP/C	Si/NO	D/EP/C				
1													
2													

Escala: D: deficiente ; EP: en proceso ; C: consolidado. Fuente: Canela, M (2023).

### Plantilla seguimiento ASISTENCIA Y PARTICIPACIÓN – semestre II

N	Apellidos	Nombres	C.I.	ASISTENCIA - PARTICIPACIÓN									
				Asesoría 1		Asesoría 2		Infografía Todas las fases		PRE SOCIALIZACION		SOCIALIZACION	
				Asistió	Estatus	Asistió	Estatus	Entregó	Estatus	Entrego SI /No	Estatus D/EP/C	Asistió SI /No	Estatus D/EP/C
				Si/NO	D/EP/C	Si/NO	D/EP/C	Si/NO	D/EP/C				
1													
2													

Escala: D: deficiente ; EP: en proceso ; C: consolidado. Fuente: Canela, M (2023).

## **CUARTO COMPONENTE - EVALUACIÓN EL CURSO**

### **ACTIVIDADES**

#### **PUNTO DE CONTROL/ PRESOCIALIZACIÓN.**



Son encuentros presenciales, para evaluar el avance del software y dominio del contexto del informe. Se realiza en la institución, con la asistencia del docente asesor, equipo de proyecto y docente jurado. Se registra calificación cuantitativa como indica la planificación. **El objeto es evidenciar el % de la adecuación de los requerimientos visto en un software de calidad.**

#### **SOCIALIZACIÓN**



Encuentro planificado con participación presencial, para **evidenciar la implantación del software (adecuación de los requerimientos)** Se realiza en la comunidad con asistencia del docente asesor, equipo de proyecto y docente jurado; se determina la aprobación de la solución. **Aprobar la UC-PST es tener una calificación > = dieciséis (16)/ 20puntos.** La misma resulta de la sumatoria de las notas obtenidas durante el trayecto.

#### **INFORME FINAL**



Entrega versión final del informe, la cual opta a una calificación cuantitativa. Determina la consistencia de los elementos que dan como resultado el desarrollo del software. Debe alcanzar una calificación mayor a 16 puntos como condición de aprobar esta evaluación.

## CUARTO COMPONENTE - EVALUACIÓN EL CURSO

### EVALUACIÓN

#### SEGUIMIENTO DEL PROYECTO.



**En el contexto de la UC-PST, se registra en plantilla como seguimiento, la evaluación cualitativa y cuantitativa de las actividades planificadas por semestre, la cual serán insumos para el cierre del trayecto, y por lo tanto evidencia para la aprobación o no del curso.**



Prof.

#### ACTA SEGUIMIENTO SEMESTRE II PROYECTO SOCIO TECNOLÓGICO

Se presenta ACTA TIENE por finalidad, PRESENTAR EVALUACIÓN CONTINUA obtenida (cuantitativa y cualitativa) hasta la FECHA \_\_\_\_\_, de las y los estudiantes de la sección \_\_\_\_ turno: \_\_\_\_\_, lapso académico \_\_\_\_\_ de la Universidad Politécnica Territorial "Juan de Jesús Montaña", en la unidad curricular PST \_\_\_\_\_, basado en la Planificación Académica. Esta se centra en el seguimiento en la elaboración del informe (FASE I, II, III, IV Y V) y la construcción e implantación del software, del proyecto titulado \_\_\_\_\_.

Código del proyecto PNFI ACA\_\_\_\_\_ / Esto sobre la base del cumpliendo de las fases de desarrollo de la metodología MERINDE. Es importante mencionar que el acta se emite por equipo de trabajo, es decir, por equipo de PST.

Nº	Apellidos	Nombre	C.L.	INFORME							
				Avance I (FASE I, II, III CONCLUIDA Y FASE IV ENCUENTRO)		Avance II (FASE I, II, III Y IV LISTO)		Avance III MANUALES		Informe Final Fase I, II, III, IV Y V - Concluidas	
				Entregó	Estratos	Entregó	Estratos	Entregó	Estratos	Entregó	Estratos
1				SU/SO	D/EP/C	SU/SO	D/EP/C	SU/SO	D/EP/C		
2											
3											

Observaciones:

Nº	Apellidos	Nombre	C.L.	SOFTWARE							
				PUNTO CONTINÚO I Modulo Seguridad – AVANCE Modulo I: CONSTRUIDO Modulo II: AVANZADO		PUNTO CONTINÚO II Modulo Seguridad/ Modulo I y III Consolidado		PUNTO SOCIALIZACIÓN producción consistente a los requerimientos y atributos de calidad		SOCIALIZACIÓN Implementación producto consistente a los requerimientos y atributos de calidad	
				Avanzó	Estratos	Avanzó	Estratos	Avanzó	Estratos	Avanzó	Estratos
1				SU/SO	D/EP/C	SU/SO	D/EP/C	SU/SO	D/EP/C	SU/SO	D/EP/C
2											
3											

Observaciones:

Nº	Apellidos	Nombre	C.L.	ASISTENCIA – PARTICIPACIÓN							
				Integración Módulo I		Integración Módulo Seguridad		Integración Presentación del software		PUNTO SOCIALIZACIÓN	
				Entregó	Estratos	Entregó	Estratos	Entregó	Estratos	participó SI/No	Estratos
1				SU/SO	D/EP/C	SU/SO	D/EP/C	SU/SO	D/EP/C	SI/No	D/EP/C
2											
3											

Escala: D: deficiente ; EP: en proceso ; C: consolidado.  
Observaciones:

Nombre	Escuela	Lugar	Firma
		DOCUMENTO	

Fuente: Canela, M (2023).

## CIERRE DEL TRAYECTO.



En este contexto, se debe llenar la plantilla de cierre de trayecto, donde se concentran todas las notas cuantitativas y cualitativas, en el orden que exige la misma. La grafica ilustra el acta de cierre del trayecto.

Elaborado por: Canela, M (2019)

**ACTA DE EVALUACIÓN FINAL  
PROYECTO SOCIO TECNOLÓGICO**

Hoy \_\_\_\_\_, la presente acta tiene por finalidad, mostrar la calificación final OBTENIDA (cuantitativa y cualitativa), de las y los estudiantes de la sección \_\_\_\_\_ turno: \_\_\_\_\_, lapso académico \_\_\_\_\_ de la Universidad Politécnica Territorial "Juan de Jesús Montilla", en la **unidad curricular PST** \_\_\_\_\_, basado en la Planificación Académica con un total del 100%, producto de la nota acumulada en el semestre I y semestre II del trayecto. Esta se centró en la implantación del proyecto titulado: \_\_\_\_\_

Código del proyecto PNFI \_\_\_\_\_. Cumpliendo las fases de desarrollo de la metodología MERINDE y la Investigación Acción Participativa Reflexiva. Es importante mencionar que el acta se emite por equipo de trabajo, es decir, por equipo de PST.



Nº	Nombres	Apellidos	C.I.	SOCIALIZACION		Auto Eva	Puntos de Control	Informe Técnico Final	Nota Final/ 100%	Nota Final /20 PTS	Firma Estudiante
				COMUNIDAD JURADO							
				20%	30%						
1											
2											
3											

Luego de plasmar las calificaciones cuantitativas de cada estudiante, se realiza el siguiente análisis, en cuanto a las condiciones de aprobación de la unidad curricular PST, las mismas establecidas los artículos 18 y 19 de la gaceta nº 39.839, resolución nº 2.593, de fecha 10 de enero de 2012

Nº	Nombres	Apellidos	C.I.	Participación con responsabilidad, compromiso, aportes durante partes de control	Asistió a Clase	Entregó Informe y aprobó el mismo		Asistió a la Socialización y aprobó la misma		Evaluadas las condiciones anteriores, se consideró aprobado el PST IV			
						Número; Algunas Vezes ; Siempre		SI	NO	SI	NO	SI	NO
1													
2													
3													

Observaciones:

**JURADO**

NOMBRE	CEDULA	CARGO	FIRMA

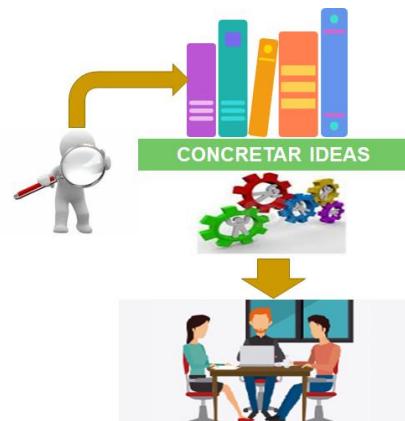
Firma y sello de la Coordinación

Fuente: Canela, M (2023).

## QUINTO COMPONENTE

### GUIA DIDÁCTICA

## BIBLIOGRAFÍA



Fuentes que permiten la construcción del conocimiento.  
La consulta del mismo, es fundamental para concretar las ideas.



## BIBLIOGRAFIA DEL CURSO

AUTOR	TÍTULO
Bruegge, B Dutoit, A (2002)	y <b>Ingeniería del Software Orientado a Objeto.</b> Primera edición, Pearson Educación. México 2002.
Booch (2004)	<b>El Lenguaje Unificado de Modelado.</b> Person Addison Wesley. España.
Buyens, J (2001)	<b>Aprenda Desarrollo de Bases de Datos web Ya.</b> Primera Edición. McGraw-Hill. España.
Cortés, D (1997)	<b>El Enfoque del Marco Lógico.</b> Instituto Universitario de Desarrollo y Cooperación. Madrid.
Cordoba, (2006).	M. Formulación y Evaluación de Proyectos. Ecoe
Crespo, M. (2009).	<b>Guía de diseño de proyectos sociales comunitarios bajo el enfoque del marco lógico.</b> Conceptos esenciales y aplicaciones. Caracas – Venezuela
Pressman, (2010).	R <b>Ingeniería de software.</b> Un enfoque práctico. Séptima Edición. McGraw-Hill. México.
Larman (2003).	<b>UML y Patrones.</b> Una introducción al análisis y diseño orientado a objeto y al proceso unificado .Segunda Edición. Person Prentice Hall. España.
Montilva otros(2004)	y <b>Sistemas de Información e Ingeniería de software.</b> Temas Selectos. Centro de Estudios en Informática. Mérida. Venezuela
Sarmiento, M. y Abreu, M. (2009).	<b>Cómo Diseñar Proyectos Comunitarios (Bajo el enfoque del Marco Lógico).</b> Guía teórico-práctica. Ministerio del Poder Popular para Ciencia, Tecnología e Industrias Intermedias. Fundacite Zulia. Venezuela
Kendall & Kendall (2005).	<b>Análisis y diseño de sistemas.</b> (6 <sup>a</sup> ed.). México: Pearson.
Marrero, C y Santos, K (2007). •	<b>Metodología de la Red Nacional de Integración y Desarrollo de Software Libre (MERINDE).</b> Centro Nacional de Tecnologías de Información. Ubicado en <a href="http://www.gnu.org/copyleft/fdl.es.html">http://www.gnu.org/copyleft/fdl.es.html</a> .



## GLOSARIO

<b>Abordaje</b>	Proceso planificado, guiado, encaminado hacia la detección de necesidades presentes en la comunidad.
<b>Artefacto</b>	Documento, gráfico, diagrama y esquemas que surgen durante el desarrollo de un software.
<b>Atributo</b>	Característica de los datos.
<b>Codificación</b>	Consiste en traducir los algoritmos en programas, usando un lenguaje de programación
<b>Comunidad</b>	Conjunto de personas que viven juntas bajo ciertas reglas o que tienen los mismos intereses.
<b>Clase</b>	Categoría de objetos con características similares.
<b>Diagnóstico</b>	Resultado que arroja luego de hacer un estudio o análisis, sobre cualquier ámbito u objeto.
<b>Diagrama</b>	Modelo de red que describe la distribución de datos en un software
<b>Entidad/Relación</b>	Poner a funcionar un software
<b>Ejecutar</b>	Disponibilidad de los recursos necesarios para llevar a cabo los objetivos o metas.
<b>Factibilidad</b>	Acción que permite al usuario la comunicación con el software.
<b>Interactiva</b>	Que se repiten
<b>Iteraciones</b>	Proceso de verificación de la adecuación de los requerimientos una vez instalado el software.
<b>Implantación</b>	Forma de ir de una pantalla a otra en un software
<b>Navegación</b>	Conjunto de artefactos que establecen la forma, estructura, navegabilidad en la adecuación de los requerimientos.
<b>Modelo del Sistema</b>	Metodología de desarrollo de software orientado a objeto.
<b>MERINDE</b>	Instancia de la clase
<b>Objeto</b>	Modelo establecido de pensar, consta de una serie de normas.
<b>Paradigma</b>	Acciones o forma de realizar alguna función
<b>Procedimiento</b>	Es un medio o aparato o sistema, que permite guiar, portar, o construir, un diseño o esquema predefinido.
<b>Plantilla</b>	Modelo de Prueba del Software
<b>Prototipo</b>	Es la planificación compuesta de actividades interrelacionadas que deben materializarse de manera coordinada para alcanzar un objetivo.
<b>Proyecto</b>	Declaraciones que identifican atributos, características, capacidades, cualidades que necesita cumplir un software para que tenga valor y utilidad para el usuario.
<b>Requerimiento</b>	Es la interpretación critica de una o varias experiencias que, a partirde su ordenamiento y reconstrucción.
<b>Sistematización</b>	Situación rutinaria
<b>Tedioso</b>	Lenguaje Unificado de Modelaje.
<b>UML</b>	

**ANEXO 1: FICHA DE REGISTRO PROYECTO.**

**LAPSO ----- UC- PST \_\_\_\_\_ SECCIÓN ----- PROFESOR: \_\_\_\_\_**

**FICHA DE PRESENTACION PROYECTO**

DATOS COMUNIDAD	
NOMBRE	
DIRECCION	
TELEFONO(S)	
PERSONA DE ENALCE CON LA COMUNIDAD	
NOMBRE	CÉDULA:
CARGO:	DEPARTAMENTO:
TELEFONO(S):	CORREO:
DATOS DEL TUTOR ACADÉMICO	
NOMBRE	CÉDULA
TELEFONO	CORREO
NOMBRE DEL SOFTWARE	

**ALCANCE DEL SOFTWARE**

MODULO A:
MODULO B:
MODULO C:
MODULO D: SEGURIDAD

**DATOS DE LOS INTEGRANTES DEL PST**

Nº	APELLIDOS	NOMBRES	C.I.	CORREO	TELEFONO
1					
2					
3					