

## **Framerwork de desarrollo de Proyectos Sociotecnológicos basado en la notación de Metamodelos de procesos de Ingeniería de Software (spem 2.0)**

Aleidys Arraiz

[aleidysarraiz@gmail.com](mailto:aleidysarraiz@gmail.com)

Universidad Politécnica Territorial de Paria “Luis Mariano Rivera”  
Venezuela

### **RESUMEN**

La Ingeniería de software constituye la base fundamental para ejecutar los procesos de desarrollo de software, se rigen por diversos métodos que utilizan una variedad de herramientas y técnicas. En Venezuela, existen propuestas metodológicas: MeRinde (Metodología de la Red Nacional de Integración y Desarrollo de Software Libre), es un modelo de proceso definido por el CNTI (Centro Nacional de Tecnologías de Información) ente adscrito al Ministerio del Poder Popular para Educación Universitaria Ciencia y Tecnología (MPPEUCYT) que desarrolla el modelo de procesos de software de acuerdo a fases, disciplinas, actividades y roles y para ello hace uso del Estándar de Modelado más conocido como lo es UML (*Unified Model Language*) para la construcción y obtención de los diferentes modelos en las distintas disciplinas del desarrollo de software. Sin embargo, UML no es la única técnica para la construcción de los modelos y existen otras notaciones que permiten modelar los procesos de una forma mucho más detallada, para promover la mejora de calidad de los productos y la mejora de su proceso de software. El objetivo principal de esta investigación fue crear un framerwork de desarrollo de Proyectos Sociotecnológicos basado en la notación de metamodelos de procesos de ingeniería de software (SPEM 2.0) adaptación del modelo de proceso MeRinde para la disciplina Requisitos y Análisis y Diseño. Para ello se contó con la Metodología de investigación de Hurtado y Toro (2005), y se recopiló información a través del instrumento del cuestionario y el enfoque de Revisión Sistemática de Kitchenham (2008).

**Descriptores:** Framework, SPEM 2.0, revisión sistemática, proyecto sociotecnológico, metamodelos.

## Framework of Sociotechnological Projects development based on Metamodelos notation of software Engineering Processes (spem 2.0)

Aleidys Arraiz

[aleidysarraiz@gmail.com](mailto:aleidysarraiz@gmail.com)

Universidad Politécnica Territorial de Paria “Luis Mariano Rivera”  
Venezuela

### ABSTRACT

Software Engineering is the fundamental basis for executing software development processes; they are governed by various methods that use a variety of tools and techniques. In Venezuela, there are methodological proposals: MeRinde (Methodology of the National Network for Integration and Development of Free Software), is a process model defined by the CNTI (National Center for Information Technologies) attached to the Ministry of Popular Power for University Education Science and Technology (MPPEUCYT) that develops the software process model according to phases, disciplines, activities and roles and for this it makes use of the Modeling Standard better known as UML (Unified Model Language) for the construction and obtaining of The different models in the different disciplines of software development. However, UML is not the only technique for the construction of the models and there are other notations that allow to model the processes in a much more detailed way, to promote the improvement of product quality and the improvement of its software process. The main objective of this research was to create a framework for the development of Sociotechnological Projects based on the notation of metamodels of software engineering processes (SPEM 2.0) adaptation of the MeRinde process model for the discipline Requirements and Analysis and Design. To that end, the research methodology of Hurtado y Toro (2005) was used, and information was collected through the questionnaire instrument and the Kitchenham Systematic Review approach (2008).

**Keywords:** Framework, SPEM 2.0, systematic review, sociotechnological project, Metamodels.

### INTRODUCCIÓN

La Ingeniería de software como rama de las ciencias de la computación constituye la base fundamental para ejecutar los procesos de desarrollo de software, que como bien es sabido se rigen por diversos métodos que utilizan una variedad de herramientas y técnicas enmarcadas dentro de los paradigmas de desarrollo de software existentes. El paradigma de desarrollo orientado a objetos representa la raíz desde la cual se originan

los procesos de desarrollo de software que se utilizan en la actualidad tal es el caso de de UP, XP, OpenUP que representan modelos de procesos inspirados en las mejores prácticas de desarrollo y además promueven la filosofía del software libre que como bien es sabido el Software Libre ha ganado seguidores a nivel mundial debido a las ventajas, tanto filosóficas como prácticas que ofrece a sus usuarios y desarrolladores. Las ventajas de este movimiento se derivan de las cuatro libertades (uso, estudio, modificación y distribución) que se fomentan en los sistemas, dando de esta forma, la posibilidad de adaptar rápidamente el software a las preferencias y necesidades tanto de usuarios como de organizaciones. Para llevar a cabo el desarrollo de software bajo las premisas señaladas anteriormente, en Venezuela por ejemplo, existen propuestas metodológicas como MeRinde (Metodología de la Red Nacional de Integración y Desarrollo de Software Libre), que es un modelo de proceso definido por el CNTI (Centro Nacional de Tecnologías de Información) ente adscrito al Ministerio del Poder Popular para la Educación Universitaria Ciencia y Tecnología (MPPEUCYT), la cual está fundamentado en las mejores prácticas para el desarrollo de software congregadas en los modelos de procesos especificados, con la finalidad de respuesta a la necesidad del Estado venezolano de gestionar proyectos de Software Libre y dar cumplimiento a la Ley de Infogobierno, acatando los estándares internacionales que permiten obtener un software de calidad. MeRinde desarrolla el modelo de procesos de software de acuerdo a fases, disciplinas, actividades y roles y para ello hace uso del Estándar de Modelado más conocido como lo es UML (*Unified Model Language*) para la construcción y/u obtención de los diferentes modelos en las distintas disciplinas del ciclo de desarrollo de software. Sin embargo, UML no es la única técnica para la construcción de los modelos y existen otras notaciones que partiendo de UML permiten modelar los procesos de una forma mucho más detallada y específica, ya que uno de los objetivos estratégicos y fundamentales en las organizaciones al momento de promover la mejora de calidad de sus productos es la mejora de su proceso de software. Para poder implantar un plan efectivo de mejora es esencial contar con un marco de trabajo (*del inglés Framework*) adecuado que permita que cada organización comprenda y establezca de forma acertada los procesos que lleva a cabo. Es necesario

entonces definir su modelo de proceso con una terminología única, semántica precisa y la gestión integrada de la medición de la organización mediante un metamodelo que sea el marco de referencia para la creación de modelos concretos.

En el Programa Nacional de Formación de Ingeniería Informática (PNFI), de la Universidad Politécnica Territorial de Paria “Luis Mariano Rivera”, (UPTP“LMR”) de Carúpano Estado Sucre, los proyectos Sociotecnológicos (PSTs) abordados desde aspectos formales de ingeniería de software, son ejecutados en la mayoría de los casos para la construcción de sistemas informáticos con la metodología MeRinde creando para cada fase del ciclo de vida del software los distintos artefactos propuestos guiados por las plantillas de diseño que este método refiere, sin embargo, los participantes del PNFI, siguen presentando dificultades para la concreción de los modelos derivados de los procesos de desarrollo, a pesar de utilizar las mencionadas plantillas, esto debido a que les cuesta formular los objetos concretos que conforman los modelos así como también seguir las actividades y tareas de una metodología tan rigurosa y fuertemente documentada como lo es MeRinde, es por ello que el objetivo principal de esta investigación es crear y presentar un framework de desarrollo de proyectos sociotecnológicos basado en la notación de metamodelos de procesos de ingeniería de software (SPEM 2.0) adaptación del modelo de proceso MeRinde para la disciplina Requisitos y Análisis y Diseño, incorporando el uso de técnicas comúnmente utilizadas, en todas las fases de desarrollo con el fin de complementar los artefactos propios de MeRinde, garantizando la calidad de los productos finales y facilitando la ejecución de los mismos por parte de los participantes del PNFI y las orientaciones de los facilitadores que realizan la función de tutores y/o asesores de los proyectos en las diferentes organizaciones de la región Pariana donde estos se realizan.

## **DESARROLLO**

Actualmente en el mercado tecnológico mundial, las exigencias de los clientes y usuarios demandan cada vez más sistemas de software complejos, eficientes y evolutivos, con un alto desempeño funcional y el preciso cumplimiento de sus objetivos de calidad, motivos por los cuales es necesario contar con modelos de proceso para el

desarrollo de software que permitan gestionar bajo un enfoque disciplinado y sistémico este tipo de proyectos. El ámbito académico no escapa de esta realidad ya que es en estos espacios donde se desarrolla el aprendizaje para la construcción de los sistemas de software (SS) bajo procesos rigurosos y especializados.

En las nuevas Universidades Politécnicas territoriales creadas en el marco de las transformaciones universitarias que vive Venezuela, se imparten los Programas Nacionales de Formación diversas áreas, tal es el caso del Programa Nacional de Formación en Informática (PNFI) cuyo objetivo fundamental es la formación de estudiantes al respecto el Documento Rector del PNFI (2008) señala:

En el marco de la prestación de servicios y generador de productos informáticos, el PNFI integra la formación, la investigación formativa o creación intelectual y la interacción socioeducativa, vinculada a la comunidad en general y bajo el principio del desarrollo del pensamiento crítico productivo e innovador mediante el trabajo colectivo en proyectos, a través de los siguientes productos y servicios: Desarrollo de soluciones informáticas y componentes de software bajo estándares de calidad, priorizando el uso de software libre, Asesoría y consultoría en el área de Informática, Instalación y ensamblaje de equipos de computación, Brindar soporte técnico a usuarios y equipos de computación, Elaboración y gestión de proyectos Informáticos, Diseñar, implementar, mantener y administrar las bases de datos. (p.8).

En la Universidad Politécnica Territorial de Paria “Luis Mariano Rivera”, de Carúpano, estado Sucre, los estudiantes del PNFI, utilizan métodos de desarrollo de software entendidos como “un conjunto de procedimientos, técnicas y ayudas para el desarrollo de productos software. En sus pasos naturales o lógicos para la construcción de software” (Stevens y Pooley, 2007,p.7), basados en las mejores prácticas y guiados por disciplinas que generan artefactos en cada una de las fases del ciclo de vida del sistema que son reflejados en documentos que contienen las actividades tareas y roles de los involucrados en dichos desarrollo.

Tal es el caso de MeRinde (Metodología de la Red Nacional de Integración y Desarrollo de Software Libre) que es la referencia utilizada en la mayoría de los PSTs ejecutados por los participantes pertenecientes al III y IV Trayecto del PNFI, esto, debido a que a pesar de tener varias alternativas al momento de seleccionar una metodología para el

desarrollo de sus proyectos prefieren seleccionar aquella que consideran más conocida y cuya documentación es guiada por plantillas prediseñadas que de una u otra forma consideran les facilitan el desarrollo que plantea MeRinde, que se encuentra estructurado por fases a saber: Inicio, Elaboración, Construcción y Transición, cada una de las cuales se realizan iteraciones, disciplinas, roles, actividades, tareas y artefactos a usar, utilizando el estándar UML( *del Inglés Unified Model Language*). Para los participantes del PNFI específicamente de los trayectos III y IV resulta complejo concebir y construir los artefactos que contienen los diagramas representados en UML para cada una de las fases que exige la metodología, por lo tanto, en la mayoría de los casos la documentación del proyecto de software es deficiente, no cumple con los Requisitos de Calidad y los participantes presentan inconvenientes tales como: Falta de claridad de las Fases, disciplinas, actividades, tareas y roles que intervienen en cada una de las fases de MeRinde. Dificultad para construir los Artefactos de la documentación incluyendo los Diagramas bajo la Notación UML. Falta de experticia en el uso de herramientas para representar los Diagramas de UML. Poca comprensión de modelos abstractos para representar los SS. Conflictos para adaptar las fases del PST al ciclo de vida de desarrollo con MeRinde. Poca consulta de los Manuales para construcción de los artefactos de Ingeniería de Software. Indagando acerca de las posibles causas que son las responsables de generar estos inconvenientes se encuentran: los bajos índices de aprobación del saber ingeniería de software, poco interés por parte de los participantes en comprender los modelos abstractos, las fases de MeRinde por ser extensas y complejas no se adaptan a la dinámica que exige el desarrollo del PST, desconocen las herramientas para la construcción de los artefactos existentes.

En la Universidad Politécnica Territorial de Paria “Luis Mariano Rivera”, de Carúpano, estado Sucre, los estudiantes del PNFI, utilizan métodos de desarrollo de software entendidos como “un conjunto de procedimientos, técnicas y ayudas para el desarrollo de productos software. En sus pasos naturales o lógicos para la construcción de software” (Stevens y Pooley, 2007,p.7), basados en las mejores prácticas y guiados por disciplinas que generan artefactos en cada una de las fases del ciclo de vida del

sistema que son reflejados en documentos que contienen las actividades tareas y roles de los involucrados en dichos desarrollo.

Tal es el caso de MeRinde (Metodología de la Red Nacional de Integración y Desarrollo de Software Libre) que es la referencia utilizada en la mayoría de los PSTs ejecutados por los participantes pertenecientes al III y IV Trayecto del PNFI, esto, debido a que a pesar de tener varias alternativas al momento de seleccionar una metodología para el desarrollo de sus proyectos prefieren seleccionar aquella que consideran más conocida y cuya documentación es guiada por plantillas prediseñadas que de una u otra forma consideran les facilitan el desarrollo que plantea MeRinde, que se encuentra estructurado por fases a saber: Inicio, Elaboración, Construcción y Transición, cada una de las cuales se realizan iteraciones, disciplinas, roles, actividades, tareas y artefactos a usar, utilizando el estándar UML( *del Inglés Unified Model Language*).

Para los participantes del PNFI específicamente de los trayectos III y IV resulta complejo concebir y construir los artefactos que contienen los diagramas representados en UML para cada una de las fases que exige la metodología, por lo tanto, en la mayoría de los casos la documentación del proyecto de software es deficiente, no cumple con los Requisitos de Calidad y los participantes presentan inconvenientes tales como: Falta de claridad de las Fases, disciplinas, actividades, tareas y roles que intervienen en cada una de las fases de MeRinde. Dificultad para construir los Artefactos de la documentación incluyendo los Diagramas bajo la Notación UML. Falta de experticia en el uso de herramientas para representar los Diagramas de UML. Poca comprensión de modelos abstractos para representar los SS. Conflictos para adaptar las fases del PST al ciclo de vida de desarrollo con MeRinde. Poca consulta de los Manuales para construcción de los artefactos de Ingeniería de Software.

Indagando acerca de las posibles causas que son las responsables de generar estos inconvenientes se encuentran: los bajos índices de aprobación del saber ingeniería de software, poco interés por parte de los participantes en comprender los modelos abstractos, las fases de MeRinde por ser extensas y complejas no se adaptan a la dinámica que exige el desarrollo del PST, desconocen las herramientas para la construcción de los artefactos existentes. En 2012, se implementó un Manual de

Ingeniería de Software, tomando en cuenta los artefactos de desarrollo de software establecidos en la metodología de la red nacional de integración y software libre MeRinde, considerando aspectos de calidad de software así como otros estándares utilizados por la Industria Nacional de Software. Sin embargo, a pesar de haber sido presentado y difundido, no fue ampliamente utilizado por todos los facilitadores de los saberes, ni tampoco consultado por la mayoría de los participantes del PNFI de los trayectos III y IV, lo que ha traído como consecuencia por ejemplo de acuerdo a datos suministrados por el departamento de control de estudios que más del 70% de la matrícula en estos trayectos no aprueba el eje de PST. Todo lo descrito anteriormente supone que cada vez es más complicado para los participantes concebir la documentación que requiere todo proyecto de ingeniería de software que son los productos derivados de los PSTs que ellos realizan específicamente en los trayectos III y IV. Esta situación tiende a retrasar la planificación académica tanto del área de saber de Ingeniería de Software de manera directa como del eje del PST ya que los participantes requieren más horas de asesorías para poder cubrir medianamente los objetivos de la metodología de desarrollo de software empleada. Es por ello que en esta investigación se plantea la creación de un framework de desarrollo de Proyectos Sociotecnológicos basado en la notación de metamodelos de procesos de ingeniería de software con el propósito de mostrar el proceso dinámico y estático de las disciplinas de MeRinde Requisitos y Análisis y Diseño con sus actividades, tareas, artefactos, técnicas y herramientas de software. Para definir y especificar el modelo de proceso asociado a los PSTs y adaptado a las Fases disciplinas tareas actividades y artefactos de MeRinde se utiliza estándar SPEM 2.0. que es un estándar de metamodelado que sirve para representar procesos de ingeniería de software, entiendo un Proceso Software (PS) como “Un conjunto coherente de políticas, estructuras organizacionales, tecnologías, procedimientos y artefactos que son necesarios para concebir, desarrollar, instalar y mantener un producto software” ( Ruiz y Verdugo, 2008. p.11 ), es así como SPEM 2.0 representa un Metamodelo que permite representar los modelos derivados de los procesos del desarrollo de software en un nivel de abstracción más sencillo que el proporcionado por la representación con UML.



## **Objetivos de la Investigación**

### **Objetivo General**

Crear un Framework de Desarrollo de Proyectos Sociotecnológicos basado en la Notación de Metamodelos de Procesos de Ingeniería de Software (SPEM 2.0).

### **Objetivos Específicos**

- ✓ Identificar los inconvenientes que presentan los participantes del PNFI en la construcción de los Artefactos derivados de las Fases de desarrollo de Software con MeRinde en los PSTs.
- ✓ Analizar las disciplinas de Requisitos y Análisis y Diseño de MeRinde para la adaptación de sus actividades a las fases de los PSTs.
- ✓ Examinar la Notación SPEM 2.0 para la identificación de las ventajas en la representación de los PS de los PST's.
- ✓ Ejecutar una Revisión Sistemática sobre las Herramientas disponibles que soportan la Notación SPEM 2.0.
- ✓ Producir un framework de desarrollo de Proyectos Sociotecnológicos basado en la notación SPEM 2.0 para las disciplinas de Requisitos y Análisis y Diseño de MeRinde.

Esta investigación orientada en la creación de un Framework de Desarrollo de Proyectos Sociotecnológicos basado en la Notación de Metamodelos de Procesos de Ingeniería de Software (SPEM 2.0), que toma como base fundamental las disciplinas de requisitos y Análisis y Diseño de la Metodología de desarrollo de software MeRinde, este marco de trabajo considera los requerimientos por los cual se rige el PNFI para ejecutar los Proyectos Sociotecnológicos específicamente en los trayectos III y IV de la UPTP "Luis Mariano Rivera", Carúpano, Estado Sucre . Con este marco de trabajo se ofrece a los participantes del PNFI poder comprender y ejecutar con mayor facilidad las fases de sus PSTs abordando las disciplinas y fases de MeRinde adaptadas y

asociadas a la estructura del PST, lo que garantiza la obtención de productos de software de mayor calidad y ejecutados en los tiempos previstos.

Es indispensable iniciar referenciando esta investigación teóricamente con el análisis de lo que representa en la Ingeniería de Software, el abordaje de los Modelos de Desarrollo aplicados para llevar a cabo el ciclo de vida de un proyecto, bien es sabido, que según Sommerville (2005) un Modelo de desarrollo abarca un proceso de desarrollo de software que tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos de los Usuarios. Este proceso es intensamente intelectual, afectado por la creatividad y juicio de las personas involucradas. Aunque un proyecto de desarrollo de software es equiparable en muchos aspectos a cualquier otro proyecto de ingeniería, en el desarrollo de software hay una serie de desafíos adicionales, relativos esencialmente a la naturaleza del producto obtenido. Modelo en cascada el primer modelo de desarrollo de software que se publicó se derivó de otros procesos de ingeniería (Royce, W, 1970). Éste toma las actividades fundamentales del proceso de especificación, desarrollo, validación y evolución y las representa como fases separadas del proceso. Desarrollo en espiral el modelo de desarrollo en espiral es actualmente uno de los más conocidos y fue propuesto por Boehm 1988. El ciclo de desarrollo se representa como una espiral, en lugar de una serie de actividades sucesivas con retrospectiva de una actividad a otra. Desarrollo incremental, Mills, H. (1980), sugirió el enfoque incremental de desarrollo como una forma de reducir la repetición del trabajo en el proceso de desarrollo y dar oportunidad de retrasar la toma de decisiones en los requisitos hasta adquirir experiencia con el sistema. Es una combinación del Modelo de Cascada y Modelo Evolutivo. Metodologías para desarrollo de software, un proceso de software detallado y completo suele denominarse "Metodología". Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, evolutivo, incremental, etc.). Adicionalmente una metodología debería definir con precisión los artefactos, roles y actividades involucrados, junto con prácticas y técnicas recomendadas, guías de adaptación de la metodología al proyecto, guías para uso de herramientas de apoyo, etc. Habitualmente se utiliza el término "método" para referirse a técnicas, notaciones y guías asociadas,

que son aplicables a una (o algunas) actividades del proceso de desarrollo, por ejemplo, suele hablarse de métodos de análisis y/o diseño. Metodologías orientadas a objetos, su historia va unida a la evolución de los lenguajes de programación orientada a objeto, los más representativos: a fines de los 60's SIMULA, a fines de los 70's Smalltalk-80, la primera versión de C++ por Bjarne Stroustrup en 1981 y actualmente Java o C# de Microsoft. A fines de los 80's comenzaron a consolidarse algunos métodos Orientadas a Objeto. En 1995 Booch y Rumbaugh proponen el Método Unificado con la ambiciosa idea de conseguir una unificación de sus métodos y notaciones, que posteriormente se reorienta a un objetivo más modesto, para dar lugar al Unified Modeling Language (UML), la notación OO continua siendo la más utilizada en la actualidad. Algunas metodologías orientadas a objetos que utilizan la notación UML son: Rational Unified Process (RUP), OPEN, MÉTRICA (que también soporta la notación estructurada). Sin embargo, la utilización de UML en las metodologías orientadas a objetos para la ejecución de los PST's ha generado dificultades en los participantes para la consecución de cada etapa, en los trabajos concebidos con MeRinde de Marrero, C y Santos, K , CNTI (2007), es un método de desarrollo de software basado en el Proceso Unificado (UP), el proceso de software propuesto por MeRinde se inspira en catorce (14) mejores prácticas, dirigidas a facilitar el desarrollo colaborativo de software entre equipos de trabajo de diversa magnitud e índole, con el fin de que se desarrolle productos de software con alta calidad, aprovechando al máximo los recursos disponibles de una forma eficaz y eficiente, con enfoques de modelos incremental e iterativos y soporta la orientación a objetos y recomienda la notación UML. La metodología utilizada MeRinde se organiza en disciplinas. Las disciplinas poseen flujos de trabajos en donde cada uno conlleva a actividades que a su vez están compuestos por un conjunto de tareas realizadas en un área determinada, las cuales tienen como objetivo producir artefactos. A su vez, en MeRinde existen actividades que se dividen en subactividades con el fin de facilitar la agrupación de tareas relacionadas. Las disciplinas que conforman esta metodología se dividirán en dos grupos. El primero comprende las disciplinas fundamentales asociadas con las aéreas de ingeniería: a. Modelado del Negocio, b. Requerimientos, c. Análisis y Diseño. Las disciplinas de MeRinde estudiadas representan el soporte principal para

abordar un proyecto de software para el marco de los PST's que se construyen en el PNFI, sus actividades, tareas y los roles que cumplen quienes las ejecutan conforman un Marco de Trabajo (*del Inglés Framework*), definiéndose *Framework* como "un conjunto de componentes físicos y lógicos estructurados de tal forma que permiten ser reutilizados en el diseño y desarrollo de nuevos sistemas de información"(Guerrero y Recaman, 2009). Un Metamodelo describe un conjunto de conceptos genéricos y sus interrelaciones, que sirven de base para la definición de Modelos de un cierto Dominio. Frameworks como alternativas didácticas para construir PST's, Afortunadamente la Didáctica y la Tecnología Educativa, consideradas éstas como el conjunto de principios y normas de enseñanza y aprendizaje con orientación hacia la práctica [Contreras, 1990], referido por Casanovas, I. (2005) han construido por acumulación y refinamiento desde hace más de medio siglo, una piedra angular de conceptos e indicadores, si bien no aplicables directamente a las tecnologías de actuales, lo suficientemente sólidos en su enunciación para ser redefinidas (o actualizables, según se prefiera) para estas nuevas . Un Framework viene a representar la acumulación de las mejores prácticas en este caso de la Ingeniería de software, que refinadas se consideran un recurso didáctico para facilitar en los participantes el desarrollo de los PST's del PNFI, considerando que los recursos si bien en los ambientes no virtuales de aprendizaje los recursos suelen ser principalmente impresos (textos) o escritos (apuntes, anotaciones en la pizarra o pizarrón), en los ambientes virtuales los recursos son digitalizados (texto, imágenes, hipertexto o multimedia). En ambos casos (presencial o virtual) se puede contar con apoyos adicionales como bibliotecas, hemerotecas, bibliotecas virtuales, sitios web, libros electrónicos, entre otros.

El marco metodológico, bien es sabido constituye la médula de la investigación de acuerdo a Hurtado y Toro (2005), se señala que "se refiere al desarrollo propiamente dicho del trabajo investigativo: la definición de la población sujeta a estudio y la selección de la muestra, diseño y aplicaciones de los instrumentos y la recolección de los datos" (p.78). Fundamentalmente se plantea el cómo se logran los objetivos propuestos, para ello se de presentar el diseño y tipo de investigación que se realiza. A continuación la matriz metodológica que da soporte a esta investigación (Ver Cuadro 1)



Cuadro 1. Matriz Metodológica de la Investigación. Arraiz 2016.

En esta etapa de la investigación se realizan los pasos necesarios para analizar y comprender toda la información que se ha recopilado, de tal manera que se clasifican, y ordenan los datos de acuerdo a como se van obteniendo, para así lograr la satisfacción de los objetivos planteados al inicio del estudio. Ver Figura 1.

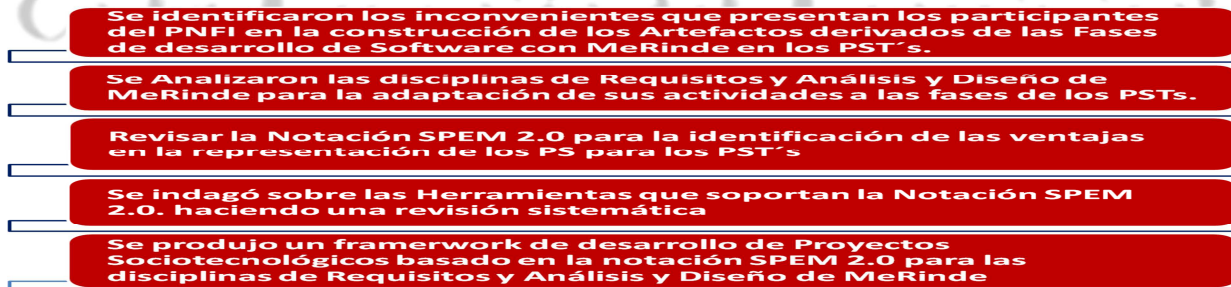


Figura 1. Presentación de Resultados (Arraiz, 2016).

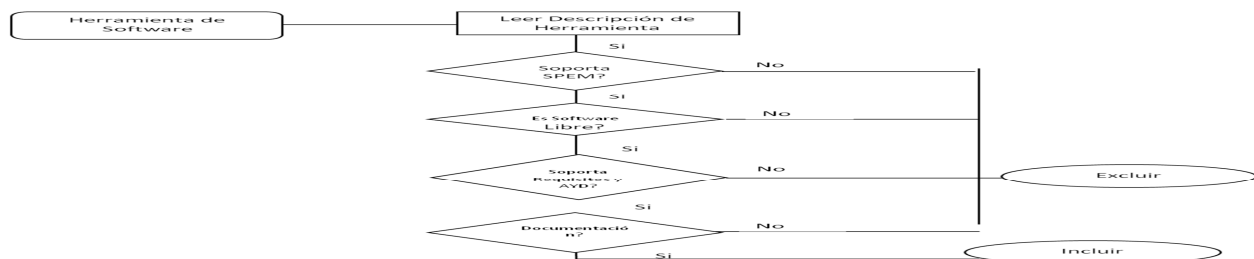
Así analizando tanto las características generales de SPEM 2.0, como sus mecanismos de trabajo se pueden especificar ventajas y desventajas asociadas a los requerimientos de los PST's, como se puede observar en los siguientes cuadros:

Cuadro 2. Aportes de SPEM al PST.

Ventajas de SPEM 2.0	Aportes en la construcción de del Proceso de Software del PST s
Provee una representación del PS estandarizada	Permite identificar las tareas básicas del Desarrollo, en especial la gestión de requisitos a través de los Casos de Uso en el PST.
Soportan el desarrollo, la gestión y el crecimiento sistemático de procesos	Permite aplicar la reutilización de métodos del proceso de desarrollo del PST, que pueden usarse también para distintos procesos.
Soportar el despliegue solo de los procesos y componentes de métodos necesarios en un determinado entorno mediante la definición de configuraciones de procesos y componentes de método	En virtud de que ningún proyecto de software exactamente igual a otro. La visión de metamodelo permite otorgar responsabilidades a los miembros del grupo de proyecto para desplegar cada parte del proceso dependiendo de su rol.
Soportar la implementación/ ejecución de procesos para proyectos en desarrollo	Permite incorporar información acerca de cómo un proceso debe ser implementado a través de la planificación del proyecto.

*Systematic Review* (en español Revisión Sistemática), consiste en una metodología empleada para la investigación, que fue desarrollada con el fin de recopilar información disponible concerniente a un tema en particular. El proceso de conducción de una investigación del tipo *Systematic Review* está muy bien definido y de forma ordenada para poder identificar, seleccionar y evaluar las investigaciones relevantes.

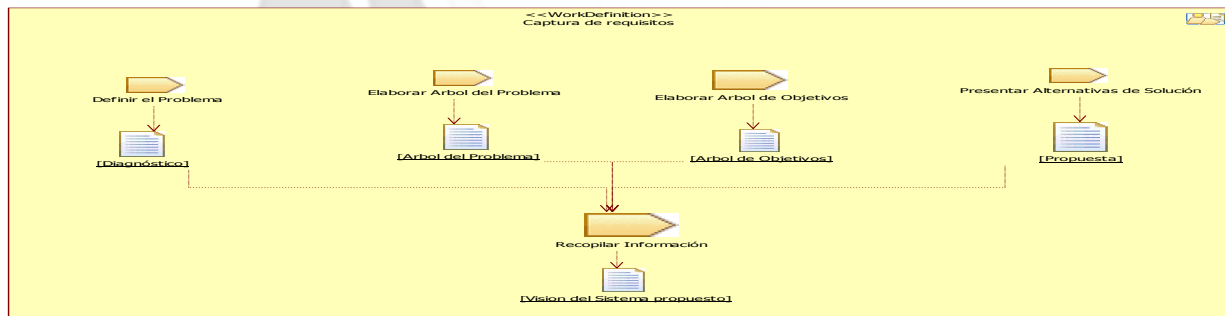
En la Comunidad del Conocimiento



En la actualidad existen variedad de técnicas enfocadas en la Ingeniería de Requisitos como disciplina que apoyan los procesos en las actividades: captura, análisis, especificación y validación de requisitos. Estas técnicas suelen estar apoyadas por herramientas de software, para su aplicación y la documentación de resultados.

Por ello en esta investigación se plantea un Framework de Desarrollo de Proyectos Sociotecnológicos basado en la Notación de Metamodelado SPEM 2.0 (FDPSPERM

V.01), que consiste en adaptar los elementos de la disciplina Requisitos del modelo de proceso MeRinde y la Disciplina de Análisis y Diseño, tomando en cuenta técnicas de Ingeniería de Software que van más allá del uso del estándar UML, indicando herramientas de Software Libre que den soporte a estas técnicas. Permitiendo mejorar los resultados para la disciplina de Requisitos en relación al flujo de trabajo, esto tiene como consecuencia la inclusión de nuevas tareas y artefactos, además de las técnicas y herramientas de software empleadas en dichas tareas.



Descripción de las Tareas de las Disciplinas de Requisitos y Análisis y Diseño adaptadas al PST.

### Especificación de Tareas de la Disciplina de Requisitos PSTs

#### Captura de Requisitos

Nombre	Definir el Problema
Rol Responsable	Analista de Sistemas
Descripción	Se identifica la situación existente en la organización objeto de estudio, extrayendo causas y consecuencias de ésta, el pronóstico de continuar la situación y la propuesta de solución.
Artefactos de Entrada	Información recabada de los involucrados (stakeholders)
Técnicas a Emplear	Entrevistas, encuestas, observación directa, revisión documental
Herramientas de Software	Editores de texto,
Artefactos de Salida	Diagnóstico

## CONCLUSIONES Y REFLEXIONES

Los procedimientos que convergen en el desarrollo de software, cada día más deben ser adaptados a las necesidades directas de los involucrados (Stakeholders), de allí que resulta indispensable personalizar los métodos que se utilizan para llevar a cabo el denominado ciclo de vida del software, por ello esta investigación permitió: La identificación de los inconvenientes que presentaban los participantes del PNFI en la construcción de los Artefactos derivados de las Fases de desarrollo de Software con MeRinde en los PSTs. Se logró a través de la recolección de información realizada con la aplicación de una encuesta, la cual evidenció las debilidades de los participantes para abordar los modelos derivados de las fases de la metodología empleada para el desarrollo de los PST's. Se examinaron las disciplinas de Requisitos y Análisis y Diseño de MeRinde y con ello se logró la adaptación de sus actividades a las fases de los PSTs, a través de un mapeo de tareas, roles y actividades. Se ejecutó una revisión exhaustiva de la Notación SPEM 2.0 focalizando de las ventajas en la representación de los PS de los PST's. Se indagó sobre las Herramientas que soportan la Notación SPEM 2.0., haciendo una revisión sistemática, que arrojó como resultados las herramientas más utilizadas para construir los metamodelos. Se diseñó y evaluó el framework de desarrollo de Proyectos Sociotecnológicos basado en la notación SPEM 2.0 para las disciplinas de Requisitos y Análisis y Diseño de MeRinde, obteniendo una alternativa consistente y coherente para soportar los procesos de desarrollo que se ejecutan en el PNFI. Es así como en el marco de las líneas de investigación del PNFI, específicamente la que abarca el área de Ingeniería de software, se presentó esta alternativa para el desarrollo de los proyectos.

## REFERENCIAS CONSULTADAS

1. Arraiz, A. (2012). *Manual de ingeniería de software para elaborar proyectos sociotecnológicos del pnfi trayecto iii y iv basado en merinde y aspectos de calidad*. Trabajo de Ascenso a Agregado. UPTP "LMR". Carúpano. Sucre. Venezuela.
2. Comisión Técnica Programa Nacional de Formación en Informática. (2008). Documento Rector Programa Nacional de Formación de Ingeniería en



- Informática. Ministerio del Poder Popular para la Educación Universitaria, Caracas, Venezuela.
3. Guerrero, C., y Recaman, H., (2009). *Marcos de Trabajo (Framework) para soportar el desarrollo de aplicaciones Web de código abierto*, 10-25, (Sic) Editorial Ltda., Bucaramanga, Colombia.
  4. Hurtado, I. y Toro J. (2005). *"Paradigmas y Métodos de Investigación en Tiempos de Cambio"*. Editorial CEC, s.a. Los Libros de El Nacional, Caracas.
  5. Kitcheman, B. (2008). *Systematic Literature Reviews in Software Engineering – A Systematic literature review Survey*.
  6. Marrero, C y Santos, K. (2007). *Metodología de la Red Nacional de Integración y Software Libre (MeRinde)*. Centro Nacional de Tecnologías de Información. Venezuela.
  7. Mills H. y O' Neil, D. (1980). *The Management of Software Engineering Part 1: Principles of software Engineering*. IBM Systems Journal. USA.
  8. Royce, W. (1970). *A Spiral Model of Software Development*. IEEE. Westcon.
  9. Ruiz y Verdugo (2008). *Ingeniería de procesos de Software*. Universidad Cantabria. Ediciones.
  10. Sommerville, I. (2005). *Ingeniería del Software*. Séptima Edición. Editorial Pearson/ Addison Wesley. España.

En la Comunión del Conocimiento