

Especificación de requerimientos con Áncora y el estándar 830

Nora Isela Zamora Aguilar¹, Ángeles Arenas Valdés¹, Karen Cortés Verdín¹, Juan Carlos Pérez Arriaga¹

¹ Universidad Veracruzana, Facultad de Estadística e Informática,
Xalapa, Ver., México

nizaguilar@hotmail.com, aarenas@uv.mx, kcortes@uv.mx, juaperez@uv.mx

Resumen. Una de las fases iniciales en la ingeniería de software es la obtención de requerimientos. Éstos son elementos medulares en el desarrollo de software, por lo cual es de gran importancia el llevar una metodología para su obtención y análisis; ya que de lo contrario, la lista de requerimientos puede ser tan errónea que causaría el fracaso total del proyecto. Se destaca la importancia e impacto de la ingeniería de requerimientos en la ingeniería de software; así como la descripción y ventajas al desarrollar las actividades metodológicamente. Se aplica la metodología Áncora para la obtención y análisis de requerimientos en un caso de estudio, y se muestra la Especificación de Requerimientos de Software (ERS) utilizando el estándar IEEE 830, complementándolo con los artefactos que se generan en la metodología utilizada.

Palabras Clave: Áncora, Requerimientos de software, IEEE 830

1 Introducción

La especificación de requerimientos permite realizar una descripción de lo que hará el software para cubrir las necesidades del usuario y garantizar así su calidad. La finalidad del presente documento es resaltar la importancia de seguir una metodología y apegarse a estándares, que aseguren el desarrollo de software de calidad. organizándolo de la siguiente manera: en la segunda sección se presentan los fundamentos de la ingeniería de software; en la tercera sección se describe la metodología Áncora y se indica en qué fase de la ingeniería de requerimientos se aplican las herramientas y artefactos de ésta metodología; en la cuarta sección se describe el estándar IEEE 830; en la quinta sección, se desarrolla la plantilla de especificación de requerimientos combinando Áncora y el estándar IEEE 830, la cual fue utilizada en un caso de estudio; por último en la sexta sección se describen los resultados obtenidos al aplicar la metodología en un caso de estudio.

2 Ingeniería de requerimientos

Un punto de referencia en la ingeniería de software es SWEBOK[1], cuya finalidad es proporcionar un compendio que muestra el conocimiento de dicha área, el cual se resume en 10 áreas de conocimiento de los conceptos básicos: Requerimientos de Software; Diseño de software; Construcción de Software; Pruebas de Software, Mantenimiento de Software; Gestión de Configuración del Software; Gestión de la Ingeniería de software; Procesos de Ingeniería de Software; Herramientas y métodos de la Ingeniería del Software y Calidad.

De todas las áreas antes mencionadas la primera corresponde a la identificación, definición, o clasificación de requerimientos, descripción de la importancia de los requerimientos cuantificables, las diferencias entre producto y proceso; y distingue entre sistema y software.

Los requerimientos describen los modelos de los procesos, actores, procesos de apoyo y gestión de procesos, calidad y mejora de procesos. El proceso de evaluación de los documentos de requerimientos permite garantizar la definición del sistema adecuado; es decir lo que el usuario espera; también se consideran prototipos y validación de los modelos, así como las pruebas de aceptación.

De acuerdo a [2], el proceso de ingeniería de requerimientos está conformado por cuatro actividades, en cada una de ellas se genera un documento, como lo muestra la figura 1.

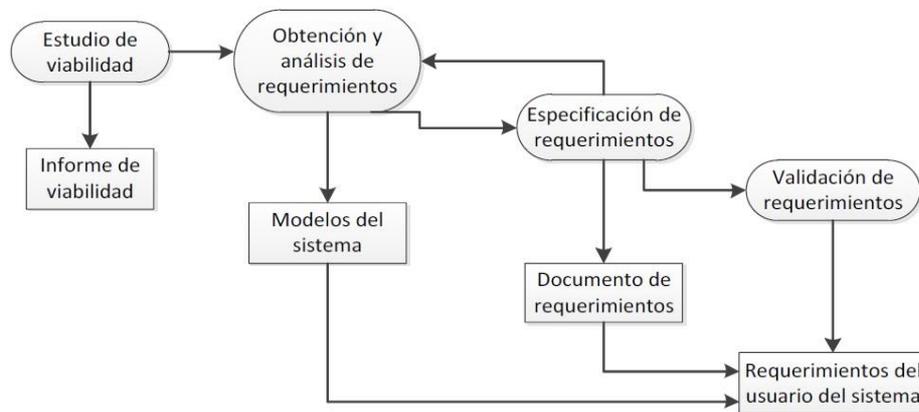


Fig. 1. Procesos de Ingeniería de requerimientos [2].

La especificación de requerimientos hace referencia a la elaboración de un documento con la finalidad de validar los requerimientos; este documento incluye los requerimientos del usuario y los requerimientos del sistema [2].

3 Áncora como metodología para la obtención de requerimientos

Áncora es una metodología que ayuda al análisis de requerimientos de software, es el acrónimo de análisis de requerimientos de software conducente al reuso de artefactos [3]. Esta metodología permite conocer claramente qué es lo que se quiere; así como el elaborar representaciones que sean claras para el usuario y poder continuar sin errores en las siguientes etapas de la ingeniería de software.

Considerando la figura 1, la cual muestra el proceso de ingeniería de requerimientos, Áncora se aplica en la segunda etapa, la cual corresponde a la obtención y análisis de requerimientos.

3.1 Ciclo de vida de Áncora

El ciclo de vida de Áncora está conformado por cinco etapas, en cada una de ellas se generan artefactos, tal como se muestra en la figura 2.

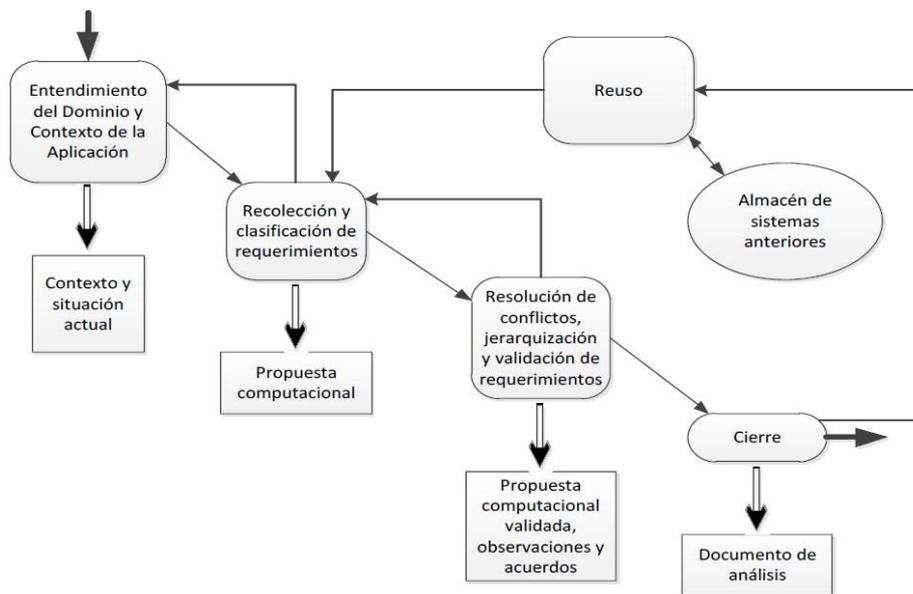


Fig. 2. Ciclo de vida de Áncora [3].

Entendimiento del Dominio y Contexto de la aplicación. El objetivo de esta etapa es identificar y aclarar las metas, dominio, actividades, situación actual del sistema y justificación del nuevo software. Las actividades que se llevan a cabo son lectura de material, entrevistas informales con el cliente, observación a usuarios, construcción de redes semánticas naturales del sistema; así como la construcción y aplicación de encuestas de actitud.

Recolección de requerimientos. En esta etapa el objetivo es desarrollar la propuesta computacional, construyendo el guión, biblioteca de desarrollo y aplicación de puntos de función.

Reuso de requerimientos. Esta etapa solo puede ser incluida si existe una base de datos con requerimientos o elementos de otros sistemas que hayan sido aprobados con anterioridad.

Resolución de conflictos, priorización y validación de requerimientos. El objetivo es resolver posibles incongruencias o conflictos entre los requerimientos identificados, asignar la prioridad a cada uno de ellos y validarlos con el usuario. Se realizan dos actividades la preparación de la reunión de reflexión y diseño, y la reunión como tal.

Cierre. Es la última fase del ciclo de vida, en ella se realizan actividades para conectar las siguientes etapas de la ingeniería de software; como el ratificar los puntos de función, envío de comunicados para los puntos pendientes y la conexión con la metodología de desarrollo seleccionada

3.2 Herramientas de Áncora

En Áncora se cuenta con diferentes herramientas, las cuales han sido tomadas de áreas distintas, como el teatro, la psicología, administración y por supuesto computación.

Las herramientas utilizadas en esta metodología son:

- Guiones y diálogos: son utilizados para la obtención de requerimientos, permitiéndole al analista tener una clara comunicación con los usuarios, así como el mostrar detalles iniciales a los diseñadores. Generando como resultado el guión de la propuesta computacional.
- Tablas y glosarios: Son utilizadas para poder reducir ambigüedades en la especificación de requerimientos por el uso del lenguaje natural; así como para aclarar términos computacionales o del área de aplicación del software.
- Modelo de datos Entidad-Relación: Permite la delimitación correcta de las fronteras del sistema [4], y el modelado de la base de datos.
- Análisis de puntos de función: Es utilizado para calcular los costos para un nuevo software, utiliza cinco indicadores para identificar cada uno de los elementos del software.
- Redes semánticas naturales: Son utilizadas para identificar los conceptos de los usuarios y eliminar las ambigüedades entre éstos.

- Técnicas de grupo nominal: Su finalidad es la toma de decisiones organizacionales para dar solución a la problemática.
- Reunión de reflexión y diseño: Tiene como objetivo obtener los modelos que permiten identificar lo que se requiere que el software realice. Se divide en cinco etapas: exploración del ambiente, análisis interno del sistema, diseño del software ideal, identificación de obstáculos y oportunidades, selección de líneas de acción y diseño de líneas de acción.

Beneficios de utilizar Áncora.

Áncora es metodología que utiliza recursos de fácil comprensión para el usuario, quien puede o no estar familiarizado con los términos técnicos del área de computación, lo que permite una participación activa. El uso de los guiones, utilizados en el teatro y literatura facilitan al usuario la comprensión del desarrollo de cada uno de los procesos, cómo éstos pueden ser implementados en un sistema automatizado y quienes son los usuarios (actores) que intervienen en ellos.

Los recursos y dinámicas empleados para la elaboración de redes semánticas permiten identificar de forma rápida aquellas acciones que son relevantes en los procesos que se implementarán en el software, también ayudan enormemente para la eliminación de ambigüedad en los conceptos involucrados en el contexto de éste.

Las reuniones de reflexión y diseño son de gran apoyo para reunir a los involucrados con el software a desarrollar, proporcionando una guía puntal para que éstas sean lo más efectivas posibles y obtener el mayor provecho de ellas sin necesidad de requerir mucho tiempo.

Todo lo anterior proporciona las herramientas necesarias para la obtención de requerimientos de forma metodológica y certera de lo que realmente necesita el usuario, ya que también facilita el cercamiento con él.

Siguiendo las fases de la metodología Áncora y generando los artefactos antes mencionado, se obtienen los requerimientos asegurando así que el software cubra todas las necesidades del usuario.

4 Estándar IEEE 830 para la especificación de requerimientos

La falta de herramientas automatizadas y poca habilidad de los ingenieros de software, en la creación de una buena especificación de requerimientos en lenguaje natural, han sido un obstáculo en el proceso eficiente de la ingeniería de requerimientos, por lo que se han desarrollado algunas propuestas, entre las que se encuentra un software Tailandés [5], NALASS [6] y herramientas que permiten la creación y verificación de especificación de requerimientos desde puntos de vista diferentes [7].

El estándar IEEE 830-1998 [8], “Recomendaciones prácticas para la especificación de requerimientos”, fue desarrollado por la IEEE y la IEEE-SA (Standards Association),

indica la estructura y organización de toda la información que debe incluirse en un buen documento de especificación de requerimientos de software. Los objetivos son: Ayudar a los clientes de software para describir con precisión lo que quieren; y a las personas para lograr: Desarrollar una estructura estándar para la especificación de requerimientos de software (ERS) en sus organizaciones; Definir el formato y contenido de las especificaciones de requerimientos de software y Desarrollar elementos de apoyo adicionales, lista de control de calidad de ERS o un manual del mismo.

La especificación de requerimientos de software obliga a los involucrados en el desarrollo del software a considerar todos los requerimientos de forma rigurosa antes de iniciar el diseño y codificación del mismo, con la finalidad de evitar el rediseño, proporcionando las bases necesarias para la estimación de tiempo y costo, referencias de verificación y validación.

En el 2011 surge el estándar ISO/IEC/IEEE 29148[9], en el cual unifica los procesos y productos involucrados en la ingeniería de requerimientos durante todo el ciclo de vida de los sistemas y software, es el resultado de la unión de los siguientes estándares: ISO/IEC 12207:2008, ISO/IEC 15288:2008, ISO/IEC/IEEE 15289:2011, ISO/IEC TR 19759, IEEE Std 830, IEEE Std 1233, IEEE Std 1362, ISO/IEC TR 24748-1, e ISO/IEC/IEEE 24765.

Es importante mencionar que uno de los puntos importantes del ISO/IEC/IEEE 29148[9], es que se mencionan los procesos y dentro de éstos las actividades que se deben llevar a cabo para una buena obtención de requerimientos, que es precisamente una de las ventajas de utilizar una metodología, en este caso *Áncora*, por lo que con los artefactos que se generan a partir de ésta, se complementa una de las plantillas propuesta por el estándar IEEE 830 y se muestra su aplicación en un caso de estudio.

5 Especificación de requerimientos de Software utilizando *Áncora* y el estándar IEEE 830

El caso de estudio analizado para la elaboración de la plantilla de especificación de requerimientos, corresponde a un software para la admisión y control escolar para el Instituto Educativo Panamericano (SIIEP-Primaria), en el nivel primaria, cuyo objetivo principal del software es automatizar los procesos de preinscripción e inscripción que actualmente realizan de forma manual y en el cual intervienen muchas personas al inicio del nuevo ciclo escolar primaria.

Como inicio de este proyecto se llevaron a cabo las actividades que permiten enlazar los requerimientos identificados con el resto de las fases y actividades de la ingeniería de software; en esta ocasión se realizó el documento de especificación de los requerimientos tomando como guía el estándar IEEE 830-1998[8] quedando la plantilla del documento con se muestra en la Tabla 1.

Es importante resaltar que el documento de Especificación de Requerimientos de Software, debe cumplir las siguientes características de calidad [10], las cuales son: Completo, Consistente, Exacto, Modificable, Clasificado, Comprobable, Trazable, Inequívoco, Válido y Verificable y que fueron consideradas al momento de elaborar el documento de especificación de requerimientos.

Tabla 1. Plantilla para la especificación de requerimientos.

Sección	Descripción	Artefactos de Ancora
1. Introducción		
1.1. Propósito	Define el propósito y a quién va dirigido el documento de especificación de requerimientos.	Entrevista
1.2. Ámbito del Sistema	Proporciona el nombre futuro del sistema, lo que hará o no, beneficios, objetivos y metas que se esperan alcanzar.	Entrevista
2. Descripción general del sistema		Red semántica natural
2.1. Contexto del sistema		Red semántica natural
2.2. Definiciones, Acrónimos y Abreviaturas	Incluye la introducción y vista general de las especificaciones.	Tablas y glosarios
2.3. Situación actual	Conceptos básicos que se utilizan.	Guión de la situación actual
2.4. Problemática	Funciones principales que actualmente se realizan.	Red semántica natural
2.5. Descripción de usuarios	Descripción del papel que desempeña cada uno de los usuarios.	Perfilado de usuario
2.6. Suposiciones y dependencias		
2.7. Restricciones	Problemática presente en el proceso actual.	
2.7.1. Entorno operativo		
2.8. Propuesta computacional		Guión de propuesta computacional
3. Descripción de requerimientos	Describe los requisitos del software que permita satisfacer los requerimientos del	
3.1. Interfaces externas		Prototipo

3.2. Requerimientos funcionales	sistema y satisfaga las necesidades del usuario. Incluye modelos que permitan dar continuidad a la ingeniería de software.	Casos de uso
3.3. Diseño de casos de uso		
3.4. Requerimientos de usabilidad		Red semántica natural y reunión de reflexión
3.5. Requerimientos de rendimiento		
3.6. Requerimientos de base de datos		Modelo de datos
3.7. Restricciones de diseño		Reunión de reflexión
3.8. Atributos del sistema		Reunión del reflexión
3.9. Otros requerimientos		Reunión del reflexión
4. Casos de prueba		Describe el plan de pruebas que se van llevar a cabo. Incluye los casos de prueba unitarios; así como los de integración.
4.1. Pruebas unitarias		
4.2. Pruebas de integración		
5. Apéndices	Información adicional que da soporte a los puntos anteriores	Entrevistas con los usuarios, redes semánticas, encuestas de actitud, gráficas, etc.

6 Resultados

La metodología de Áncora fue aplicada en el desarrollo software SIIEP-Primaria del Instituto Educativo Panamericano S. C., siguiendo todas las actividades necesarias, utilizando las herramientas propias de la metodología para generar los siguientes artefactos:

- Red semántica natural
- Encuesta de actitud
- Guión de la situación actual
- Guión de la propuesta computacional
- Perfilado de usuarios
- Casos de uso
- Modelo de datos

- Prototipo
- Especificación de usabilidad
- Cálculo de Puntos por casos de uso
- Planteamiento de casos de prueba

Se realizaron tres reuniones de Reflexión y Diseño de las cuales se obtuvo la priorización de requerimientos, se establecieron aspectos técnicos para la plataforma de desarrollo y algunas características no funcionales pero de importancia para el usuario final.

Con toda esta información fue posible generar un Documento con la especificación de los Requerimientos de Software (ERS) que cumplía las características mencionadas en la sección cinco (Tabla 2). Dicho documento permitió la fácil transición a las siguientes fases de desarrollo, ya que el sistema estaba bien delimitado, con los requerimientos claramente especificados, un conjunto de datos bien definidos en un modelo preliminar de datos y el prototipo que cubría en su totalidad los casos de uso. Todo lo anterior fue aprobado en una reunión de Reflexión y diseño, por lo tanto solo quedaba colocar el código correspondiente a cada uno de los eventos que conformaban el prototipo.

Tabla 2. Características del documento de ERS.

Característica	Validación realizada
Correcto	Cada requerimiento fue revisado y aceptado por los involucrados con el software en las reuniones de reflexión. Cada requerimiento tiene una descripción de lo que debe cumplir, definidas las entradas y salidas; y la relación con otros requerimientos
Sin ambigüedades	Cada requerimiento se apega a lo establecido en el proceso y los conceptos empleados fueron unificados con en los glosarios, acrónimos y abreviaturas
Completo	El documento contiene toda la información necesaria para comprender las necesidades del cliente y pasar a las siguientes fases de desarrollo.
Consistente	Los requerimientos son revisados en las reuniones de reflexión y diseñados para asegurar que los requerimientos no se contradigan Cada requerimiento tiene una descripción de lo que debe cumplir, definidas las entradas y salidas; y la relación con otros requerimientos
Importante y/o estable	Cada requerimiento fue priorizados por los usuarios en las reuniones de reflexión y agrupados de acuerdo a la funcionalidad y secuencia establecida por el proceso
Verificable	Cada uno de los requerimientos se incluyeron en los casos

	de uso, los cuales fueron revisados y aceptados por los usuarios desde la definición hasta su funcionalidad validada con la planificación y ejecución de pruebas
Modificable	Cada requerimiento tiene un identificador y se describe cada uno por separado para referirse a él se manera no ambigua
Trazable	Cada requerimiento es trazable a las siguientes fases, por ejemplo a los casos de prueba

7 Conclusiones y trabajos futuros

El documento muestra la importancia de la ingeniería de requerimientos para desarrollar correctamente un software que satisfaga las necesidades del usuario.

Describe las fases de la metodología Áncora, la cual proporciona herramientas y artefactos que son de gran utilidad en la definición de requerimientos.

También se presenta la estructura del documento de definición de requerimientos, la cual se elaboró tomando como guía el estándar IEEE 830-1998, complementándolo con diversos artefactos de Áncora; la estructura diseñada fue utilizada como el documento en el que se formalizaron los requerimientos del usuario y que sirvieron como base para dar continuidad a la ingeniería en el desarrollo del sistema SIIEP-Primaria.

El utilizar Áncora y el estándar IEEE830, permitieron el acercamiento y correcta comunicación con el usuario (siendo esta una prioridad para la metodología), facilitando la definición, claridad y priorización de los requerimientos; así como la identificación y establecimientos de procesos previos y necesarios antes de automatizar un proceso.

La aportación principal de este documento es mostrar la importancia de la ingeniería de requerimientos utilizando metodologías y estándares que facilitan la comunicación con el usuario; quien al estar involucrado y comprender la importancia de realizar todo el proceso de forma metodológica lo que da valor a cada una de las fases a desarrollar; así como unificar los conceptos utilizados en la organización.

Al concluir el presente documento, se tienen como trabajos futuros:

- El análisis de herramientas automatizadas para la especificación de requerimientos en lenguaje natural, que permita agilizar la elaboración del documento
- Elaboración de la plantilla de especificación de requerimientos, incorporando aspectos para la unificación de procesos y producto.

Referencias

1. A. Abran, J W. Moore, P. Bourque, & R.Dupuis, (Edits), Guide to the Software Engineering Body Knowledge, Published by IEEE, Los Alamitos, (2004)
2. I. Sommerville, Ingeniería de Software, Séptima edición, Pearson Educación, S. A., Madrid España, (2006)
3. Sumano, Ma, de los Ángeles, Áncora: Análisis de requerimientos de software conducente al reuso de artefactos, Universidad Veracruzana, Xalapa, Ver., México, (2006)
4. Kendall, K. E., & Kendall, J. E., Análisis y diseño de sistemas, 6° edición, Pearson Educación de México, México, (2005)
5. Thongglin, K., Cardey, S., & Greenfield, P., Thai Software Requirements Specification Pattern. SoMeT 2013, 12th IEEE International Conference on Intelligent Software Methodologies, Tools and Techniques, (2003)
6. Georgiades, M., & Andreou, A., Automatic Generation of a Software Requirements Specification (SRS) Document. 10th International Conference on Intelligent Systems Design and Applications, (2010)
7. Matsuo, Y., Ogasawara, K., & Ohnishi, A., Software Requirements Specifications. 978-1-4244-4840-1/101 ©2010 IEEE, (2010)
8. IEEE, Institute of Electrical and Electronics Engineers, Especificación de Requerimientos según el estándar de IEEE-830-1998, Recuperado el 03 01, 2013, desde Institute of Electrical and Electronics Engineers: www.ieee.org, (1998)
9. ISO/IEC 2011, IEEE2011, International Standard ISO/IEC/IEEE 29148, Systems and software engineering – Life cycle processes – Requirements engineering, IEEE Standars Activities Department, Piscataway, NJ, USA, (2011)
10. INKtopia Limited., TECHwhirl. Recuperado el 13 de Julio de 2014, de Helping the world, explorer the art of content and communications: <http://techwhirl.com/writing-software-requirements-specifications/>, (2014)