



Conference: Congreso Interdisciplinario de Energías Renovables -
Mantenimiento Industrial - Mecatrónica e Informática

Booklets



RENIECYT

Registro Nacional de Instituciones y
Empresas Científicas y Tecnológicas

CONACYT

1702902

RENIECYT - LATINDEX - Research Gate - DULCINEA - CLASE - Sudoc - HISPANA - SHERPA UNIVERSIA - E-Revistas - Google Scholar
DOI - REDIB - Mendeley - DIALNET - ROAD - ORCID

Title: METODOLOGÍA PARA EL DESARROLLO DE SOFTWARE
PROPIO DE LA UPFIM

Author: Omar, REYNA-ANGELES, Zaila, HERNÁNDEZ-TAPIA, Susana Leticia,
SOTO-FERNÁNDEZ, Marcos Yamir, GÓMEZ RAMOS

Editorial label ECORFAN: 607-8534
BCIERMMI Control Number: 2018-03
BCIERMMI Classification (2018): 251018-0301

Pages: 14
RNA: 03-2010-032610115700-14

ECORFAN-México, S.C.

244 – 2 Itzopan Street
La Florida, Ecatepec Municipality
Mexico State, 55120 Zipcode
Phone: +52 1 55 6159 2296
Skype: ecorfan-mexico.s.c.
E-mail: contacto@ecorfan.org
Facebook: ECORFAN-México S. C.

Twitter: @EcorfanC

www.ecorfan.org

Holdings

Mexico	Colombia	Guatemala
Bolivia	Cameroon	Democratic
Spain	El Salvador	Republic
Ecuador	Taiwan	of Congo
Peru	Paraguay	Nicaragua

CONTENIDO

- I. INTRODUCCIÓN
- II. PLANTEAMIENTO DEL PROBLEMA Y OBJETIVO
- III. ESTADO DEL ARTE
- IV. ELEMENTOS GENERALES DE LA MADSP
- V. ETAPAS DE LA MADSP
- VI. RESULTADOS Y TRABAJO A FUTURO
- VII. REFERENCIAS

I. Introducción

- La subdirección de sistemas informáticos de la UPFIM se encarga de la administración de las TIC`s de la institución
- Una de las responsabilidades es el desarrollo y mantenimiento de los sistemas informáticos
- Debido al poco personal con el que cuentan solo les permite administrar el sistema de control escolar

II. Planteamiento del Problema y Objetivo

- Existe poca automatización de procesos
- Se propone que docentes y alumnos del programa educativo de ISC participen en desarrollo de software
- Es necesaria una metodología para el desarrollo de los proyectos de software que considere etapas, actividades, pasos a seguir, responsabilidades, documentación a generar y nomenclatura de los documentos.
- De no hacerlo no se contaría con procedimientos a seguir en el desarrollo de los sistemas y su documentación, si los desarrolladores tienen experiencia podrían lograr aplicaciones funcionales pero escasamente documentadas, lo que ocasionaría una gran incertidumbre a futuro

II. Planteamiento del Problema y Objetivo

- Diseñar una metodología actual e innovadora apropiada a las necesidades organizacionales y funcionales de dos áreas: la subdirección de sistemas informáticos y de la dirección del programa educativo de Ingeniería en Sistemas Computacionales de la UPFIM, la metodología permitirá estandarizar el desarrollo de sistemas y optimizar tiempos de desarrollo ya que se tendrán los procesos bien definidos con los controles adecuados para el seguimiento y cumplimiento de las actividades del equipo de desarrollo.
- El contar con esa metodología permitirá generar una documentación técnica correcta y completa, para lograr un control de cambios de software apropiado e integro

III. Estado del Arte

Metodologías Tradicionales y RUP	Metodologías Ágiles
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo	Basadas en heurísticas provenientes de prácticas de producción de código
Cierta resistencia a los cambios	Especialmente preparados para cambios durante el proyecto
Impuestas externamente	Impuestas internamente (por el equipo de desarrollo)
Proceso mucho más controlado, con numerosas políticas/normas	Proceso menos controlado, con pocos principios
Requiere extensa documentación para controlar todo el flujo del desarrollo	Poca documentación, el control es mínimo pues se basan en la disciplina y capacidad de los desarrolladores.
Documenta todo para tener un buen control de versiones y cambios una vez ya liberado el sistema.	No se generan documentos a menos que sean muy importantes, para esta filosofía la documentación no es esencial.
Existe un contrato prefijado	No existe contrato tradicional o al menos es bastante flexible
El cliente interactúa con el equipo de desarrollo mediante reuniones	El cliente es parte del equipo de desarrollo
Grupos grandes y posiblemente distribuidos	Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio
Más artefactos	Pocos artefactos
Más roles	Pocos roles
La arquitectura del software es esencial y se expresa mediante modelos	Menos énfasis en la arquitectura del software

IV. Elementos Generales de la MADSP

Tipo Ágil

- a) Preparados para cambios durante el proyecto
- b) No existe contrato tradicional con el cliente, porque no es necesario
- c) Pocos roles
- d) Grupos pequeños trabajando en el mismo sitio
- e) Buena comunicación entre el grupo de desarrollo
- f) Gran capacidad y experiencia en el grupo de desarrollo
- g) El cliente es parte del grupo de desarrollo y está en el sitio de trabajo
- h) Basada en la producción de aplicaciones funcionales lo más pronto posible, considerando solo la documentación esencial para control de cambios de versiones a futuro
- i) Poco control, ya que se basa en la disciplina del grupo de desarrollo.

IV. Elementos Generales de la MADSP

RUP

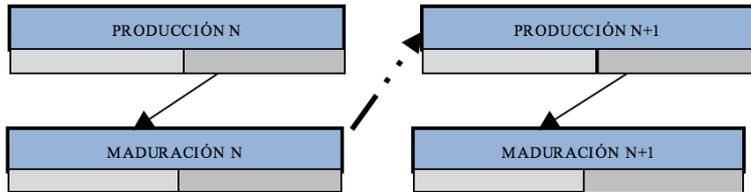
- a) Los artefactos a utilizar como entradas y salidas de las etapas son algunos de los que se utilizan en esta metodología.
- b) Los controles que se llevarán a cabo aunque son mínimos están basados en los propuestos por RUP

IV. Elementos Generales de la MADSP

En cuanto al grupo de trabajo se recomienda que estén en un mismo espacio físico para lograr una mejor comunicación y se trabaje en parejas como recomienda las metodologías ágiles en grupos pequeños para desarrollo. Se deben considerar los roles de líder, desarrollador, cliente y consultor, todos ellos son parte de la UPFIM el líder y desarrolladores deberá ser elementos de la SSI o de la ISC, el cliente es del área solicitante de la aplicación y el consultor es alguien también de la UPFIM que conozca de la normativa, reglamentos o leyes que deben considerarse en los procesos a automatizar.

V. Etapas del MADSP

Si durante la Producción, Maduración o Entrega ocurre algún cambio en los requerimientos del cliente, en el diseño, o en la integración de módulos se puede integrar la modificación pero eso alterara el plan de desarrollo inicial, y es justificable solo si el responsable del cambio es el cliente debido a un cambio en alguna normativa que afecte el proceso a automatizar.



Los módulos del software en la etapa Producción pueden trabajarse en serie o en paralelo, al igual que en la Maduración; la decisión dependerá del número de desarrolladores disponibles. Cada incremento esta dado por la finalización de la

- Flujo necesario 
- Flujo en caso de cambios en Concepción 
- Flujo en caso de cambios en Producción 
- Flujo en caso de cambios en Maduración 

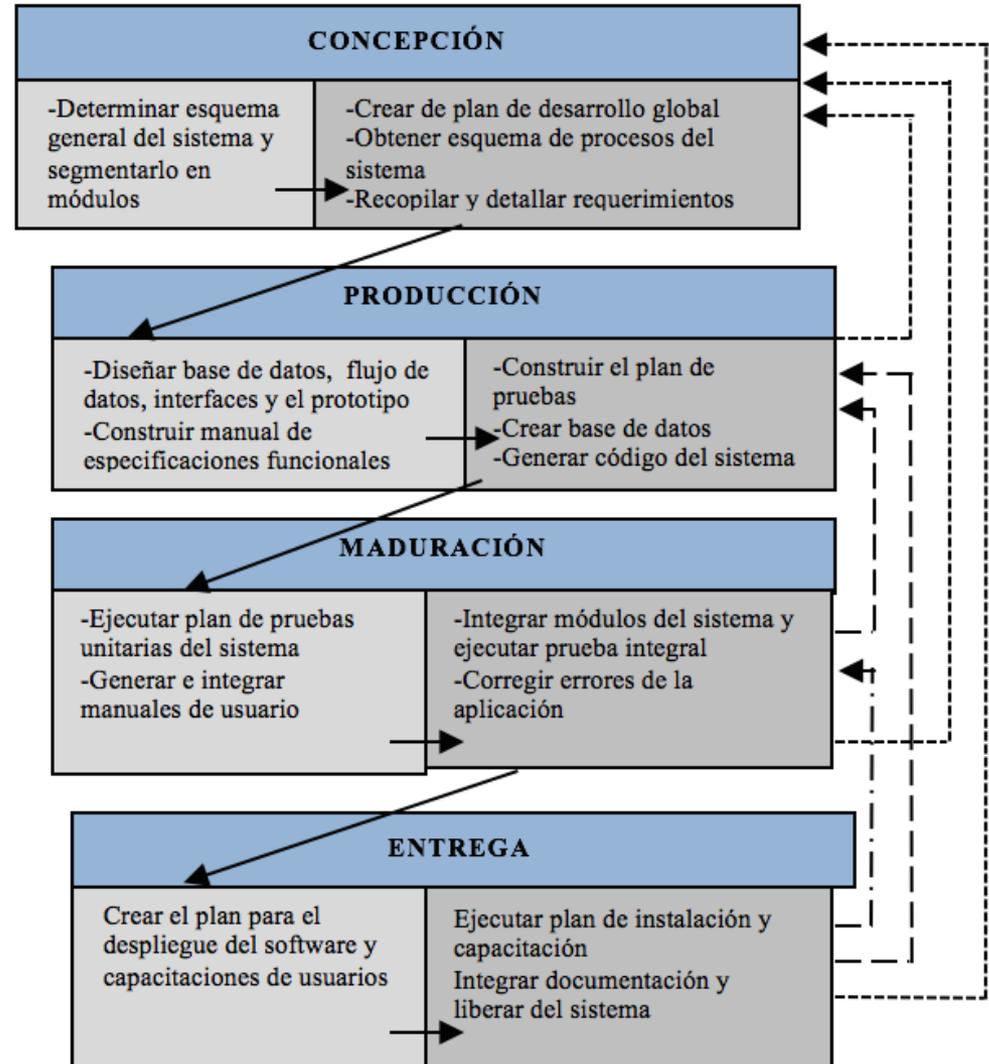


Figura 1 Diagrama de las etapas de MADSP

Fuente: *Elaboración propia*

VI. Resultados y trabajo a futuro

Tarea	Objetivo
1. Realizar un análisis de la organización del equipo de desarrollo de la UPFIM, para determinar los requerimientos que deberá cubrir la metodología que se propondrá	Conocer al detalle el área de trabajo y obtener el perfil de los recursos humanos (desarrolladores de la SSI, desarrolladores de la ISC y del subdirector de sistemas informáticos)
2. Hacer un estudio profundo de las metodologías de desarrollo de software Robustas y Ágiles para determinar los métodos útiles a la propuesta que realizará.	<p>Conocer y entender a profundidad las metodologías útiles a la propuesta que se realizará.</p> <p>Determinar que se van a tomar de las metodologías conocidas, para armar la propuesta.</p>
3. Definir las generalidades de metodología a partir de los principios, filosofías y fases de otras metodologías existentes, considerando el contexto del grupo de desarrollo de la UPFIM y las necesidades de la SSI.	<p>Generar la propuesta de la MADSP para proponerla a la SSI de la UPFIM con el fin de contar con una metodología actual e innovadora apropiada a sus necesidades organizacionales y funcionales</p>
4. Establecer las etapas de la metodología, las actividades a realizar en cada una de ellas, el flujo de trabajo y comunicación entre las actividades de cada etapa y la secuencia de las mismas.	

Tabla 2. Actividades para definir la metodología de desarrollo de la SSI de la UPFIM

Fuente: *Elaboración propia*

VI. Resultados y trabajo a futuro

Trabajo a Futuro	Objetivo
1. Definir para cada etapa descripción y formato de los artefactos (registros) de cada actividad, estos pueden ser impresos o electrónicos y pueden referirse a documentos o código de aplicaciones, así como el responsable de generarlos y resguardarlos	Contar con elementos de control durante el desarrollo y documentar las distintas actividades y etapas para los futuros mantenimientos a los sistemas y control de versiones
2. Establecer una nomenclatura para cada artefacto	Identificar de manera estandarizada a cada registro de las actividades, etapas y sistemas para un mejor control de la documentación y apoyo al mantenimiento y control de versiones de las distintas aplicaciones desarrolladas
3. Desarrollar aplicación para la administración de producción académica, científica y tecnológica del personal docente de la UPFIM	Aplicar la metodología en un caso de desarrollo de software para evaluar su eficacia y analizar resultados obtenidos

Tabla 3. *Trabajos a futuro*

Fuente: *Elaboración propia*

VII. Referencias

- Bernd Bruegge, Dutoit Allen H.(2002); Ingeniería de Software Orientado Objetos(1era ed.); México; Prentice Hall
- Concepción Suárez Ramiro(2007); Metodología de Gestión de Proyectos en las Administraciones Públicas según ISO 10.006; Tesis; Universidad de Oviedo; (Disponible en: http://www.tdr.cesca.es/TDX/TDR_UOV/TESIS/AVAILABLE/TDR-0215108-122610//UOV0024TRCS.pdf Consultado: el 28 de Diciembre del 2017)
- Diego Berrueta(2006); Programación Extrema y Software Libre; Artículo; (Disponible en: http://www.asturlinux.org/archivos/jornadas2006/ponencias/ProgExtrema_Berrueta/ponencia-sl-y-xp.pdf Consultado: el 23 de diciembre del 2017)
- Gerardo Fernández Escribano(2002); Introducción a Extreme Programming; Artículo; (Disponible en; <http://www.dsi.uclm.es/asignaturas/42551/trabajosAnteriores/Trabajo-XP.pdf> Consultado: el 22 de diciembre del 2017)
- Jacobson Ivar, Booch Grandy, Rumbaugh James(2004); El Proceso Unificado de Desarrollo de Software(1era ed.); México; Pearson Education
- Julio César Rueda Chacón(2006); Aplicación de la Metodología Rup para el Desarrollo Rápido de Aplicaciones Basado en el Estándar J2EE; Tesis; Universidad de San Carlos de Guatemala; (Disponible en: <ftp://ftp.itmerida.mx/.../Documentacion%20RUP/Tesis%20Aplicacion%20RUP.pdf> Consultado: el 2 de enero del 2018)

VII. Referencias

- Martin Fowler(2003); The New Methodology; Artículo; (Disponible en; <http://www.programacionextrema.org/articulos/newMethodology.es.html>. Consultado: el 20 de diciembre del 2017)
- Martínez Alejandro, Martínez Raúl (2006); Guía a Rational Unified Process; Escuela Politécnica Superior de Albacete; Artículo; (Disponible en: <http://www.dsi.uclm.es/asignaturas/42551/trabajosAnteriores/Trabajo-Guia%20RUP.pdf> Consultado: el 13 de diciembre del 2017)
- Peláez Sánchez José Ignacio (2004), Introducción a la Ingeniería de Software; Universidad de Málaga; (Disponible en: http://www.lcc.uma.es/~jignacio/index_archivos/TEMA1.pdf. Consultado: el 7 de diciembre del 2017)
- Peláez Sánchez José Ignacio (2004), Gestión de Proyectos de Software; Universidad de Málaga; (Disponible en: http://www.lcc.uma.es/~jignacio/index_archivos/TEMA2.pdf. Consultado: el 7 de diciembre del 2017)
- Peláez Sánchez José Ignacio(2004), Comunicación en el Desarrollo de Sistemas; Universidad de Málaga; (Disponible en: http://www.lcc.uma.es/~jignacio/index_archivos/TEMA3.pdf. Consultado: el 9 de diciembre del 2017)

VII. Referencias

- Peláez Sánchez José Ignacio (2004), Metodologías para el Desarrollo de Software; Universidad de Málaga; (Disponible en: http://www.lcc.uma.es/~jignacio/index_archivos/TEMA4.pdf. Consultado: el 12 de diciembre del 2017)
- Pressman. Roger S.(2002); Ingeniería de Software(5ta ed.); México; Mc Graw Hill
- Roxana Giandini, Gabriela Pérez, Claudia Pons (2009); Un lenguaje de Transformación específico para Modelos de Proceso del Negocio; Artículo; Universidad Abierta Interamericana de Buenos Aires; Universidad Nacional de La Plata; (Disponible en: <http://imgbiblio.vaneduc.edu.ar/fulltext/files/TC104933.pdf>; Consultado el 25 de Mayo del 2018)
- Schenone Marcelo Hernán; Diseño de una Metodología Ágil de Desarrollo de Software; Tesis; Universidad de Buenos Aires, Facultad de Ingeniería; 2004; (Disponible en: <http://materias.fi.uba.ar/7500/schenone-tesisdegradoingenieriainformatica.pdf>; Consultado: el 21 de diciembre del 2017)
- Stephen A. White, PHD Derek Miers; BPMN Guía de Referencia y Modelado Edición Digital; 2009; (Disponible en: http://www.futstrat.com/books/BPMN_edicion_espanol.php; Consultado el 16 de mayo del 2018)



ECORFAN®

© ECORFAN-Mexico, S.C.

No part of this document covered by the Federal Copyright Law may be reproduced, transmitted or used in any form or medium, whether graphic, electronic or mechanical, including but not limited to the following: Citations in articles and comments Bibliographical, compilation of radio or electronic journalistic data. For the effects of articles 13, 162,163 fraction I, 164 fraction I, 168, 169,209 fraction III and other relative of the Federal Law of Copyright. Violations: Be forced to prosecute under Mexican copyright law. The use of general descriptive names, registered names, trademarks, in this publication do not imply, uniformly in the absence of a specific statement, that such names are exempt from the relevant protector in laws and regulations of Mexico and therefore free for General use of the international scientific community. BCIERMMI is part of the media of ECORFAN-Mexico, S.C., E: 94-443.F: 008- (www.ecorfan.org/ booklets)