

**“IMPLEMENTACIÓN DE UNA METODOLOGÍA DE
DESARROLLO ÁGIL PARA EL ANÁLISIS, DISEÑO E
IMPLEMENTACIÓN DEL SISTEMA SOFTWARE Y
APLICACIÓN DE UN CASO DE ESTUDIO PRÁCTICO, EN
EL DEPARTAMENTO DE TECNOLOGÍAS DE
INFORMACIÓN Y COMUNICACIONES DE LA ESCUELA
TÉCNICA DE LA FUERZA AÉREA”**

Luis Egas
Juan Játiva

Ing. De Software

UFA-EPEL 2014



ÍNDICE

- **Capítulo 1.- Definición del Problema**
 - Del Departamento TIC de la ETFA
 - Objetivos
 - Hipótesis
- **Capítulo 2.- Marco Teórico**
 - Antecedentes Históricos
 - Antecedentes Conceptuales y Referenciales
 - Antecedentes Contextuales
- **Capítulo 3.- Determinación de la Metodología Adaptable al Departamento TIC**
 - Selección de las Metodologías a ser investigadas
 - Estudio comparativo de las metodologías seleccionadas

ÍNDICE

- Capítulo 4.- Desarrollo de la metodología adaptable y aplicación del caso de estudio práctico
 - Ciclo de Vida de la Metodología Adaptable “XP TIC’S”
 - Implementación de la Metodología Adaptable “XP TIC’S”

INTRODUCCIÓN



La Ingeniería de software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, y el estudio de estos enfoques.



El proceso de la ingeniería de software requiere llevar a cabo numerosas tareas agrupadas en etapas, al conjunto de estas etapas se le denomina ciclo de vida.



Un objetivo de décadas ha sido el encontrar metodologías, modelos, paradigmas y filosofías de desarrollo, que sean sistemáticas, predecibles y repetibles, a fin de mejorar la productividad en el desarrollo y la calidad del producto software.

CAPÍTULO I.- Definición del Problema

- En el presente capítulo se describe parte de la investigación, donde se da a conocer los problemas que ha generado, el hecho de no utilizar metodologías para el desarrollo de un sistema software y cómo la oportuna aplicación de estas en el proceso de ingeniería del software nos sirve para efectivizar el propósito deseado, mediante la aplicación de un caso de estudio práctico, como es la satisfacción del cliente.



Del Departamento TIC

Una de las mayores deficiencias en la práctica de construcción de software es la **poca atención** que se presta al seguimiento de **parámetros y prácticas** específicas durante todas las etapas de desarrollo de software.

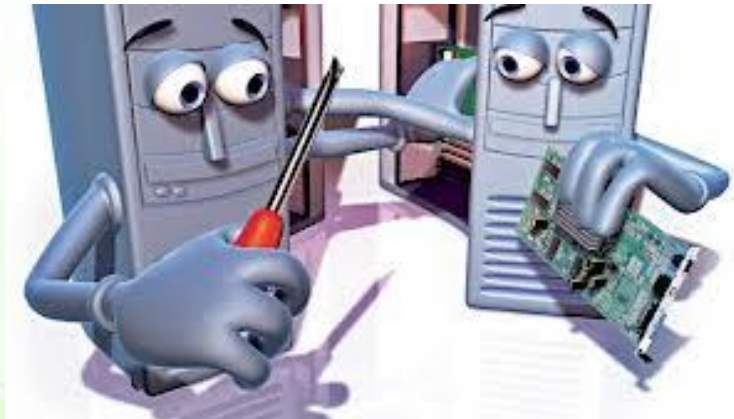
La **falta** de aplicación de **herramientas, normas, estándares y metodologías** disponibles que proporcionen un soporte tecnológico, conceptual y humano en el Departamento TIC, ocasiona la **adquisición de sistemas** a otras empresas por la deficiente calidad de software diseñado y tiempos tardíos de entrega.

La **ausencia de la comunicación** entre los desarrolladores y el Jefe de Departamento, junto al **pobre levantamiento de requisitos iniciales** y la carencia de documentación formal genera: el **retraso** del desarrollo normal del producto software, la **sobrecarga** de trabajo, la **baja calidad** y la **entrega tardía** del producto software.

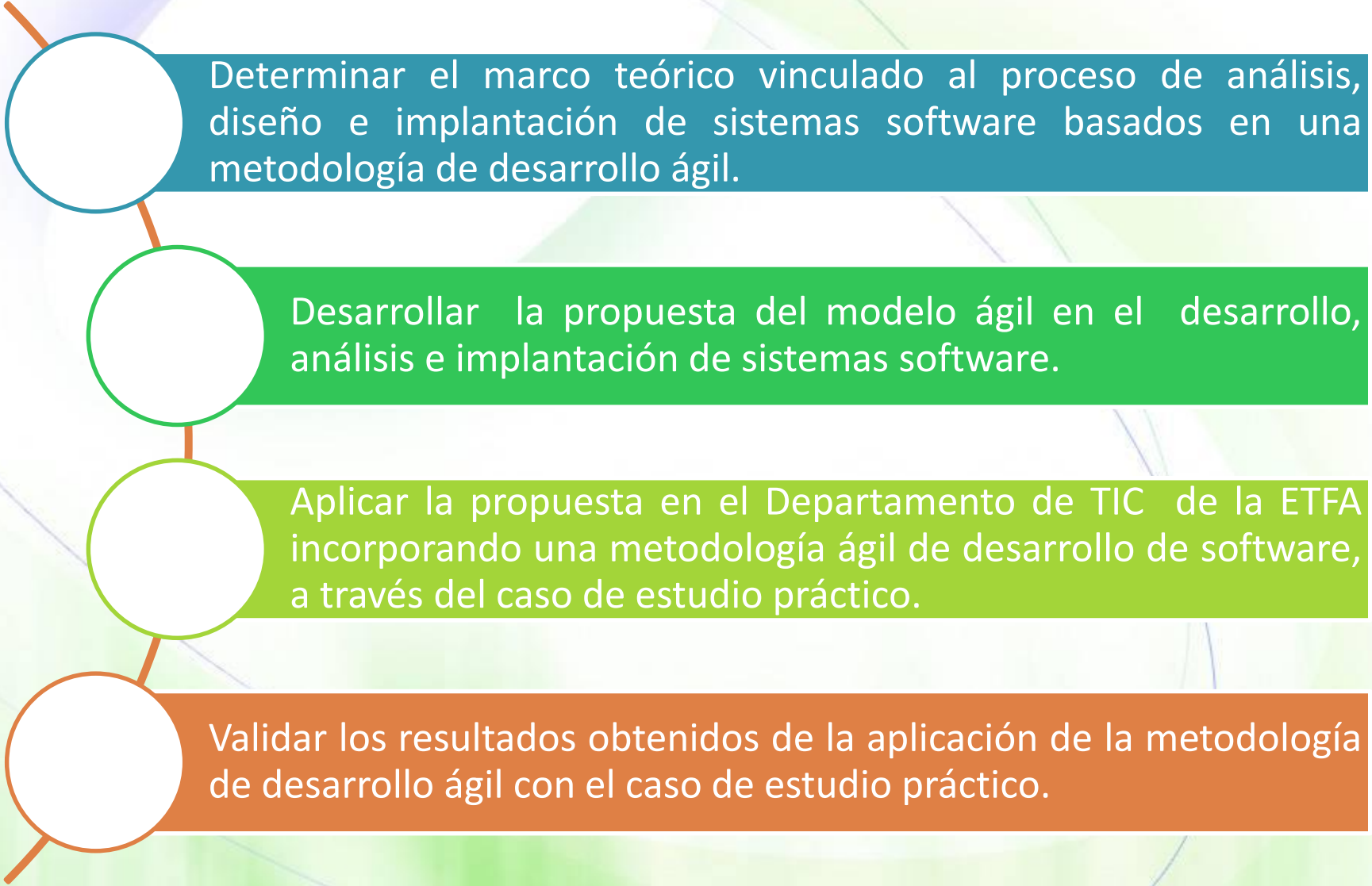


Objetivo General

- Implementar una metodología de desarrollo ágil que optimice el análisis, diseño e implantación de sistemas software, con la aplicación del caso de estudio práctico “Sistema Inteligente para el Control Disciplinario de los Aspirantes a Tropa de la ETFA”, en el Departamento de Tecnologías de la Información y Comunicaciones de la Escuela Técnica de la Fuerza Aérea.



Objetivos Específicos



Determinar el marco teórico vinculado al proceso de análisis, diseño e implantación de sistemas software basados en una metodología de desarrollo ágil.

Desarrollar la propuesta del modelo ágil en el desarrollo, análisis e implantación de sistemas software.

Aplicar la propuesta en el Departamento de TIC de la ETFA incorporando una metodología ágil de desarrollo de software, a través del caso de estudio práctico.

Validar los resultados obtenidos de la aplicación de la metodología de desarrollo ágil con el caso de estudio práctico.

Hipótesis

- Si se implementa una metodología de desarrollo ágil y se aplica a un caso de estudio práctico “Sistema Inteligente para el Control Disciplinario de los Aspirantes a Tropa”, entonces se optimiza el análisis, diseño e implementación del sistema software, en el Departamento de Tecnologías de Información y Comunicaciones de la Escuela Técnica de la Fuerza Aérea de la ciudad de Latacunga.



CAPÍTULO 2.- Marco Teórico

Antecedentes Históricos

- Evolución de las Metodologías de Desarrollo de la Ingeniería de Software en el Proceso la Ingeniería de Sistemas Software.

Antecedentes Conceptuales y Referenciales

- Caracterización Tecnológica del Proceso de la Ingeniería del Software
- Caracterización Tecnológica de las Metodologías de Desarrollo de la Ingeniería de Software

Antecedentes Contextuales

- Justificación de la problemática

Antecedentes Históricos

Primera Etapa Cronológica (1950-1960)

“Programación o técnicas de codificación”

- El desarrollo estaba a cargo de los programadores.
- No existían metodologías de desarrollo. Los programadores se limitaban a la **codificación**. Hay **inconformidad** del usuario.
- Se necesitaba métodos sencillos. Entonces se crearon los **lenguajes de programación de tercera generación**, que diferían de las generaciones anteriores (lenguaje ensamblador conocido como segunda generación y lenguaje de máquina como primera generación) en que sus instrucciones eran de alto nivel (comprensibles por el programador, como si fueran lenguajes naturales) e independientes de la máquina.
- FORTRAN, COBOL.

Segunda Etapa Cronológica (1960-1970)

“Modelo de procesos”

- Se empieza el desarrollo de modelos simples, preocupados por los **procesos**, mas no por los datos.
- Surge el modelo de procesos “**Code and Fix**”; siendo la respuesta al carecer de una estructura formal a seguir.
- Este modelo implementaba el código y luego se pensaba en los requisitos, diseño, validación y mantenimiento.
- Aparece la **Crisis del Software**: excesos de costos, escasa fiabilidad, insatisfacción de los usuarios y tiempos tardíos de entrega del SW.

Antecedentes Históricos

Tercera Etapa Cronológica (1970-1985)

“Proceso de Desarrollo
Software y Modelos
Tradicionales del Ciclo
de Vida”

- El término **Ingeniería del Software** aparece como respuesta de la crisis.
- Se empezó a tomar la importancia de los **datos**, y para solucionar sistemas complejos empezó el **análisis por partes o etapas**, se introducen la **planeación y la administración** del SW.
- Aparece el **Ciclo de Vida de Desarrollo de Software** como un consenso para la construcción centralizada de software.
- Para formalizar la estructura del ciclo de vida se logra establecer el **Proceso de Desarrollo Software**.
- Surgen **Modelos Tradicionales de ciclo de vida SW**.
- **Modelo en Cascada** (como respuesta al modelo de procesos), Modelo en V, Modelo Iterativo (prototipos), Modelo de Desarrollo Incremental (cascada + prototipos).

Cuarta Etapa Cronológica (1985-1999):

“Métodos Rápidos e
inicios del Desarrollo
Ágil de la Ingeniería de
Software”

- Surge **Modelo Espiral** (evolución a través de iteraciones) como enfoque evolutivo, donde se estima esfuerzo y tiempo.
- Se conceptualiza y formaliza el **Método** de la ingeniería de software.
- Surge el desarrollo SW de **Métodos Rápidos** (prototipo + funcionalidad iterativa), los cuales reducen tiempo del ciclo de vida SW, acelerando el desarrollo (rápido=adaptarse a cambios).
- Estos surgen por: la inestabilidad del entorno técnico y el cliente es incapaz de definir requisitos.
- Aparece **RAD**, a la necesidad de entregar sistemas muy rápido, en respuesta al desarrollo tradicional.
- Posterior **DSDM, SCRUM, XP, RUP**.

Antecedentes Históricos

Quinta Etapa Cronológica (2000 al presente) “Metodologías del Proceso de la Ingeniería de Software”

- Creciente demanda de software en organizaciones pequeñas.
- Necesidad software de bajo costo.
- Crecimiento de métodos más simples y rápidos que desarrollarán software funcional.
- Se forma la **Alianza Ágil**, que promueve el desarrollo ágil de aplicaciones.
- Se introduce las **Metodologías de Desarrollo Ágil**, basados en el desarrollo iterativo e incremental, teniendo presente los cambios y respondiendo a estos mediante la colaboración de un grupo de desarrolladores **auto-organizados y multidisciplinares**.
- Scrum, Crystal Clear, XP, ASD, FDD, DSDM, etc.
- La evolución ha desembocado hasta conformar la concepción de las **Metodologías del Proceso de Ingeniería de Software**, la cual describe el conjunto de herramientas, técnicas, procedimientos y soporte documental para el diseño del **Sistema Software**.



Antecedentes Contextuales y Referenciales

Caracterización Tecnológica del Proceso de la Ingeniería del Software

- Proceso
- Ingeniería
- Software
- Procesos de Ingeniería del Software

Etapas de Procesos de Ingeniería del Software

- Análisis de requisitos
- Especificación de requisitos
- Arquitectura
- Programación
- Pruebas
- Documentación
- Mantenimiento

Tipos de Procesos de Ingeniería del Software

- Procesos informales
- Procesos gestionados
- Procesos metodológicos
- Procesos de mejora

Entidades Reguladoras del Proceso de Ingeniería del Software

- Estándares IEEE
- Normas ISO/IEC

Antecedentes Contextuales y Referenciales

Caracterización Tecnológica de las Metodologías de Desarrollo de la Ingeniería de Software

- Método
- Metodología
- Desarrollo
- Metodología de Desarrollo Software

Clasificación de las Metodologías de Desarrollo

- **Metodologías Tradicionales:** Modelo en Cascada, en V, Iterativo, de Desarrollo Incremental, Espiral, RAD, etc.
- **Metodologías Ágiles:** RUP, Scrum, Familia Crystal, XP, ASD, FDD, DSDM, LSD, etc.

METODOLOGÍAS ÁGILES	METODOLOGÍAS TRADICIONALES
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

Antecedentes Contextuales

Retrasos en las entregas de software

Sobrecarga de trabajo

Insatisfacción del cliente

Falta de personal y recursos necesarios

Carencia de un marco de trabajo, estándares, métodos y/o metodologías a seguir

Adicional, según los resultados el 100% del personal acuerdo con la creación de un marco de trabajo a través del uso de métodos y/o metodologías

Según los resultados, el 80% del personal del Dpto. TIC afirma que existen dificultades que han ido ocurriendo durante el proceso de desarrollo software



CAPÍTULO 3.- Determinación de la Metodología

- En la presente investigación, se realizará un análisis y estudio completo de todas las metodologías del proceso de ingeniería de software. Partiendo del estudio de las metodologías ágiles, la contrastación con las metodologías tradicionales y la selección de la o las metodologías que mejor se adaptan al Dpto. TIC, mediante el uso de herramientas y frameworks de investigación.



Selección de las Metodologías

- Estudio de las Metodologías Ágiles relevantes:
 - La metodología con mayor presencia en Internet.
 - La metodología mejor documentada.
 - Metodologías certificadas y con training.
 - Metodologías con comunidades.

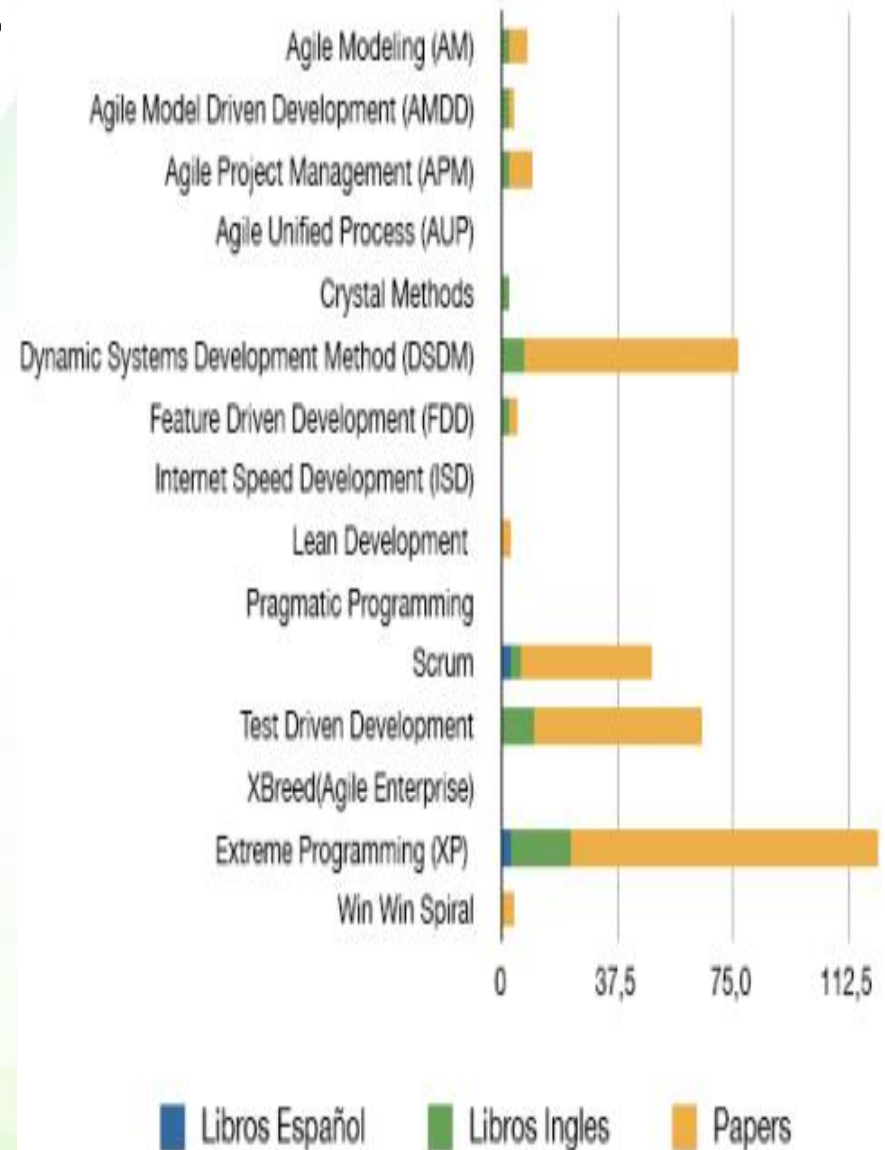


Selección de las Metodologías

• La metodología mejor documentada

Las cinco metodologías mejor documentadas son:

1. Extreme Programming (XP)
2. Feature Driven Development (FDD)
3. Dynamic System Development Method (DSDM)
4. Scrum
5. Adaptative Software Development (ASD)



Selección de las Metodologías

● Metodologías con Certificación

- Dynamic System Development Method tiene un DSDM Certified.
- Feature Driven Development tiene FDD Certified.
- Scrum dispone de Scrum Certified.

● Metodologías con Training

- Agile Modeling.
- Agile Model Driven Development.
- Adaptative Software Development.
- Crystal methods.
- Dynamic System Development Method.
- Feature Driven Development.
- Lean Development.
- Scrum.
- Extreme Programing.

Selección de las Metodologías

● Metodologías con Comunidades

a) Metodologías asociadas a la Agile Alliance:

- Agile Modeling.
- Agile Model Driven Development.
- Crystal methods.
- Dynamic System Development Method.
- Feature Driven Development.
- Lean Development.
- Pragmatic Programming.
- Scrum.
- Test Driven Development.
- Extreme Programming.



b) Metodologías con comunidades o alianzas diferentes:

- Adaptive Software Development, con Declaration of Interdependence for modern management.
- Dynamic System Development Method, con DSDM Consortium.
- Scrum, con Scrumalliance.

Selección de las Metodologías

- El objetivo inicial era seleccionar un máximo de cinco metodologías, pero en vista de los resultados estas han sido las metodologías seleccionadas:
 1. Adaptative Software Development (ASD).
 2. Crystal Methods.
 3. Dynamic Sytems Development Method (DSDM).
 4. Scrum.
 5. Feature Driven Development (FDD).
 6. Extreme Programming (XP).

Estudio Comparativo de las Metodologías

- Primera Parte: Formulario Orientación tradicional vs Orientación ágil

ORIENTACIÓN ÁGL		ORIENTACIÓN TRADICIONAL	
VALOR	IMPORTANCIA	VALOR	IMPORTANCIA
Individuo y las interacciones del equipo	3	El proceso y las Herramientas	2
Desarrollar software que funciona	3	Conseguir una buena Documentación	2
Colaboración con el Cliente	2	Negociación Contractual	2
Respuesta al cambio	3	Seguimiento de un Plan	2
MEDIA	2,75		2

De acuerdo a los resultados arrojados, se demuestra que se sobrevalora lo indicado por los valores del manifiesto ágil, con una orientación ágil de media 2,75 mayor a una orientación tradicional de media 2.

Estudio Comparativo de las Metodologías

● Segunda Parte: Formulario Cumplimiento principios ágiles

	Principios del Manifiesto Ágil	Prioridad
1	La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.	2
2	Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.	3
3	Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.	2
4	La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.	2
5	Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo	3
6	El diálogo cara a cara es el método más efectivo para comunicar información dentro de un equipo de desarrollo.	3

	Principios del Manifiesto Ágil	Prioridad
7	El software que funciona es la medida principal de progreso.	3
8	Los procesos ágiles promueven un desarrollo sostenible. Los promotores, los desarrolladores y usuarios deberían ser capaces de mantener una paz constante	3
9	La atención continua a la calidad técnica y al buen diseño mejora la agilidad.	3
10	La simplicidad es esencial.	3
11	Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.	2
12	En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.	2
	TOTAL	31/36

Según el resultado obtenido, se deduce que las metas según el enfoque del equipo de desarrollo del Departamento TIC se orientan en un 86,11% al cumplimiento íntegro o total de los principios ágiles.

Estudio Comparativo de las Metodologías

● Tercera Parte: Framework Iacovelli

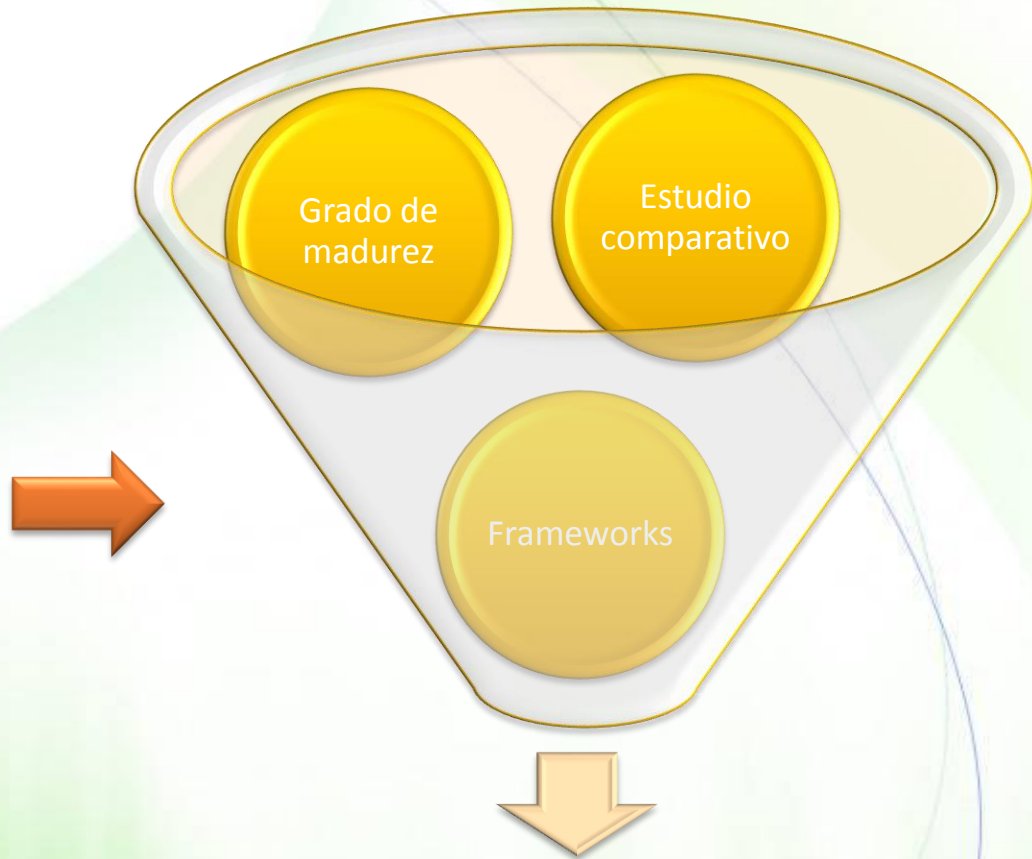
		METODOLOGÍAS ÁGILES						
		ASD	CRISTAL	DSDM	XP	FDD	SCRUM	
USO	¿Por qué utilizar una metodología ágil?	Respeto de las fechas de entrega	1	1	1	0	1	1
		Cumplimiento de los requisitos	1	1	1	1	1	1
		Respeto al nivel de calidad	1	0	0	0	0	0
		Satisfacción del usuario final	0	0	1	0	0	0
		Entornos turbulentos	0	0	1	1	0	1
		Favorable al Off shoring	0	0	0	1	1	0
		Aumento de la productividad	0	0	0	1	0	0
CAPACIDAD DE AGILIDAD	¿Cuál es la parte de agilidad incluida en la metodología?	Iteraciones cortas	0	0	0	1	1	1
		Colaboración	1	1	1	1	0	0
		Centrado en las personas	1	1	1	1	0	0
		Refactoring político	1	1	0	0	1	1
		Prueba político	0	1	0	0	1	0
		Integración de los cambios	1	0	1	1	0	1
		De peso ligero	0	0	0	1	1	1
		Los requisitos funcionales pueden cambiar	1	0	1	1	0	1
		Los requisitos no Funcionales pueden cambiar	1	0	0	0	0	0
		El plan de trabajo puede cambiar	0	0	0	1	0	0
		Los recursos humanos pueden cambiar	1	1	0	1	0	0
		Cambiar los indicadores	0	0	0	1	0	0
		Reactividad (AL COMIENZO DEL PROYECTO, CADA ETAPA, CADA ITERACIÓN)	0	0	1	1	0	0
Intercambio de conocimientos (BAJO, ALTO)	0	0	1	0	1	1		

			METODOLOGÍAS ÁGILES					
			ASD	CRISTAL	DSDM	XP	FDD	SCRUM
APLICABILIDAD	¿Cuándo un ambiente es favorable para usar esta metodología?	Tamaño del proyecto (PEQUEÑO, GRANDE)	0	0	0	1	0	0
		La complejidad del proyecto (BAJA, ALTA)	0	0	0	1	0	0
		Los riesgos del proyecto (BAJO, ALTO)	0	0	0	1	0	0
		El tamaño del equipo (PEQUEÑO, GRANDE)	0	1	0	1	0	0
		El grado de interacción con el cliente (BAJA, ALTA)	1	0	1	1	0	0
		Grado de interacción con los usuarios finales (BAJA, ALTA)	0	0	1	0	0	0
		Grado de interacción entre los miembros del equipo (BAJA, ALTA)	0	1	1	1	0	0
		Grado de integración de la novedad (BAJA, ALTA)	1	0	1	1	0	0
		La organización del equipo(AUTO-ORGANIZACIÓN, ORGANIZACIÓN JERÁRQUICA)	1	0	1	0	1	0
		PROCESOS Y PRODUCTOS	¿Cómo están Caracterizados los procesos de la metodología?	Nivel de abstracción de las normas y directrices				
Gestión de proyectos	1			1	1	0	1	1
Descripción de procesos	1			1	1	1	1	0
Normas y orientaciones concretas sobre las actividades y Productos	0			0	0	0	0	1
Las actividades cubiertas por el método ágil								
Puesta en marcha del proyecto	0			0	1	0	0	0
Definición de requisitos	1			0	1	1	1	1
Modelado	1			1	1	1	1	1
Código	1			1	1	1	1	1
Pruebas unitarias	1			1	1	1	1	1
Pruebas de integración	1			1	1	1	1	1
Prueba del sistema	1			1	1	1	1	1
Prueba de aceptación	1			0	0	0	0	0
Control de calidad	1			0	0	0	1	0
Sistema de uso	0			0	0	0	0	0
Productos de las actividades del método ágil								
Modelos de diseño	0			0	1	0	1	1
Comentario del código fuente	1			1	1	1	1	1
Ejecutable	1			1	1	1	1	1
Pruebas unitarias	1			1	1	1	1	1
Pruebas de integración	1			1	1	1	1	1
Pruebas de sistema	1			1	0	1	1	1
Pruebas de aceptación	1			0	1	0	0	0
Informes de calidad	1	0	0	0	0	0		
Documentación de Usuario	0	0	1	0	0	0		
TOTAL			30	21	31	33	24	23

Estudio Comparativo de las Metodologías

Cuarta Parte: Estado actual de las metodologías (grado de madurez)

Metodología	Estado Actual
ASD	En construcción
Crystal Methods	En construcción/ Activa
DSDM	Activa
SCRUM	Activa
FDD	Activa
XP	Activa



*Metodología **Extreme Programming (XP)**

*Podría ajustarse la DSDM, ASP, Scrum (ponderaciones altas)

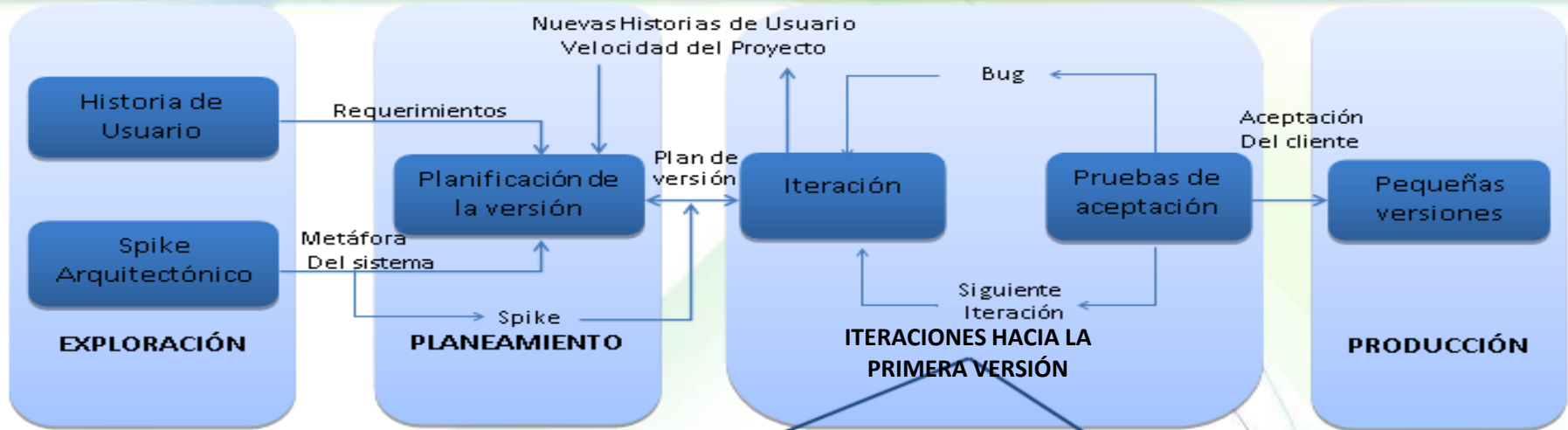
*Combinación de metodologías.

CAPÍTULO 4.- Desarrollo de la metodología adaptable y aplicación del caso de estudio práctico.

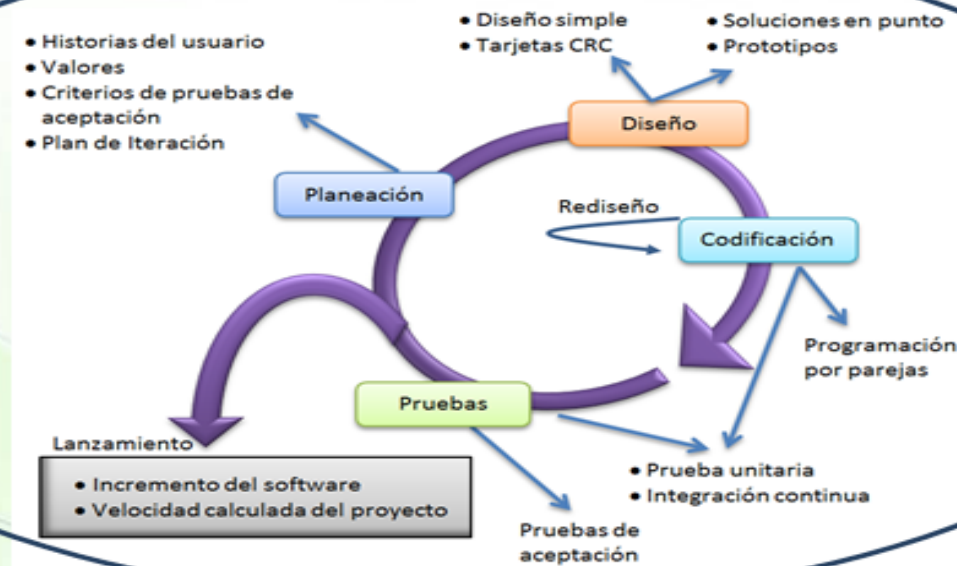
- En base al estudio de selección previamente realizado, se desarrollará la metodología ágil adaptable al Dpto. TIC.
- Se establecerá del ciclo de vida adaptable.
- Se establecerá el proceso de desarrollo SW.
- Se aplicará la metodología con un caso de estudio práctico.



Ciclo de Vida de la Metodología Adaptable "XP TIC'S"



1. Fase de Exploración
2. Fase de Planeación del Proyecto
3. Fase de Iteraciones
4. Fase de Producción
5. Fase de Mantenimiento
6. Fase de Muerte del Proyecto



DESCRIPCIÓN DE LA METODOLOGÍA DE TRABAJO

- **INTRODUCCIÓN**
 - **PROPÓSITO DE ESTE DOCUMENTO**
 - **ALCANCE**
- **DESCRIPCIÓN GENERAL DE LA METODOLOGÍA**
 - **FUNDAMENTACIÓN**
 - **VALORES DE TRABAJO**
- **PERSONAS Y ROLES DEL PROYECTO**
- **FASES DE LA METODOLOGÍA SELECCIONADA “XP TIC’S”.**

1ª Fase: Exploración. (Metodología)

- **RECOLECCIÓN DE REQUERIMIENTOS GENERALES**

Nombre	Departamento	Requerimiento/Historia Usuario	Prioridad estimada
Persona1	Dpto. "A" ETFA	H1.	Alta/Media/Baja
Persona2	Dpto. "B" ETFA	H2.	Alta/Media/Baja
....			

1ª Fase: Exploración. (Práctica)

● RECOLECCIÓN DE REQUERIMIENTOS GENERALES

Nombre	Departamento	Requerimiento/Historia Usuario	Prioridad estimada
Aspirantes	Personal	Gestión de Aspirantes	Alta
Instructores	Personal	Gestión de instructores	Media
Faltas	Dpto. Cuerpo Atros.	Gestión de faltas	Alta
Sanciones	Dpto. Cuerpo Atros.	Gestión de sanciones	Alta
Perfiles	Dpto. Cuerpo Atros.	Perfiles de Usuario	Media
Registro	Dpto. Cuerpo Atros.	Gestión de sanciones Aspirantes	Media
Registro	Dpto. Cuerpo Atros.	Gestión de méritos Aspirantes	Media
Reportes	Dpto. Cuerpo Atros.	Gestión de reportes	Media

1ª Fase: Exploración. (Metodología)

● PRINCIPALES PROCESOS A GENERARSE

No.	Nombre del Proceso	Descripción del Proceso	Actor	Funciones del Actor
1	Proceso/Gestión A	Procesos lógicos, Interrelaciones, funcionalidades, etc.	Actor A	Función A
...				

1ª Fase: Exploración. (Práctica)

● PRINCIPALES PROCESOS A GENERARSE

N o .	Nombre del Proceso	Descripción del Proceso	Actor	Funciones del Actor
1	Gestión de Aspirantes	El sistema permitirá dar de alta, baja y modificaciones de los Aspirantes.	Administrador	Alta, baja, modificaciones y búsquedas.
2	Gestión de instructores	El sistema permitirá dar de alta, baja y modificaciones de los instructores.	Administrador	Alta, baja y modificaciones.
3	Gestión de faltas	El sistema permitirá dar de alta y modificaciones de las faltas.	Administrador	Alta y modificaciones
4	Gestión de Sanciones	El sistema permitirá dar de alta y modificaciones de las sanciones.	Administrador	Alta y modificaciones
5	Gestión de perfiles de usuario	Usuario administrador, regulador y secretario.	Sistema	Perfiles con permisos.
6	Gestión de sanciones de Aspirantes	Registro de sanciones	Administrador, Regulador, Secretario	Registro
7	Gestión de méritos de Aspirantes	Registro de méritos	Administrador, Regulador, Secretario	Registro
8	Gestión de reportes	Reportes por Aspirante, fecha, promoción.	Administrador, Regulador, Secretario	Reportes.

1ª Fase: Exploración. (Metodología)

- **ESTIMACIÓN GENERAL DEL PROYECTO**

HISTORIA	TIEMPO ESTIMADO
Historia 1	1 a 8 horas
Historia 2	1 a 8 horas
...	
Tiempo Estimado Total	Σ horas

1ª Fase: Exploración. (Práctica)

● ESTIMACIÓN GENERAL DEL PROYECTO

HISTORIA	TIEMPO ESTIMADO HORAS
Gestión de Aspirantes	125
Gestión de instructores	125
Gestión de faltas	125
Gestión de sanciones	50
Perfiles de Usuario	25
Registro sanciones	110
Registro de méritos	25
Gestión de reportes	100
Tiempo Estimado Total Horas	685

2ª Fase: Planificación del Proyecto . (Metodología)

● HISTORIA DE USUARIOS

Historia de Usuario	
Número:	Usuario:
Nombre historia:	
Prioridad en negocio: (Alta / Media / Baja)	Riesgo en desarrollo: (Alta / Media / Baja)
Puntos estimados:	Iteración asignada:
Programador responsable:	
Descripción:	
Observaciones:	

2ª Fase: Planificación del Proyecto . (Práctica)

HISTORIA DE USUARIOS

Historia de Usuario	
Número: 1	Usuario: Desarrollo
Nombre historia: Alta de Aspirantes	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 5	Iteración asignada: 1
Programador responsable: Xavier Játiva – Luis Egas	
Descripción: <ul style="list-style-type: none">• Ingreso de Alumnos con los siguientes atributos: nombres, apellidos, cedula, dirección, teléfono, teléfono móvil, edad, fecha de nacimiento, lugar de nacimiento, mail, promoción.• Los alumnos de primer año dispondrán de 120 méritos.• Los alumnos de segundo año tendrán 80 méritos.	
Observaciones: Los méritos son la cantidad limite de puntos que posee un aspirante durante un año según el Reglamento de Sanciones y Recompensas para Aspirantes a Oficiales y Tropa de las FFAA; los puntos se van disminuyendo mediante son sancionados.	

2ª Fase: Planificación del Proyecto . (Metodología)

● **PRIORIDAD**

Actividad N°	Descripción	Prioridad
1		
2		
3		
...		

2ª Fase: Planificación del Proyecto . (Práctica)

PRIORIDAD

Historia N°	Descripción	Prioridad	Iteración
1	Alta de Aspirantes	Alta	1
2	Modificación de Aspirantes	Baja	3
3	Baja de Aspirantes	Baja	3
4	Alta de Instructores	Alta	1
5	Modificación Instructores	Baja	3
6	Baja de Instructores	Baja	3
7	Alta de Faltas	Alta	1
8	Modificación de Faltas	Alta	1
9	Alta de Sanciones	Alta	1
10	Modificación de Sanciones	Alta	1
11	Perfiles de Usuario	Alta	1
12	Registro de sanciones leves y graves	Media	2
13	Registro de sanciones atentatorias	Media	2
14	Notificación de sancionados	Baja	3
15	Notificación mensual de deméritos	Media	2
16	Sugerencia de regulación de faltas graves	Media	2
17	Reporte por Promoción	Media	4
18	Reporte Aspirante	Media	4
19	Reporte de todos los Alumnos	Media	4

3ª Fase: Iteraciones. (Metodología)

● DIVISIÓN EN ITERACIONES

Primera Iteracion	H1, H2, H3
Segunda Iteracion	H4, H5, H6
Tercera Iteracion	H7, H8, ...
....	Hn.....

3ª Fase: Iteraciones. (Práctica)

● DIVISIÓN EN ITERACIONES

Iteración	Historias	Fecha Inicio	Fecha Reunión	Fecha Entrega
Primera Iteración	H1, H4, H7, H8, H9, H10, H11	09-09-2013	23-09-2013	18-10-2013
Segunda Iteración	H12, H13, H15, H16	21-10-2013	29-10-2013	26-11-2013
Tercera Iteración	H2, H3, H5, H6 H14	27-11-2013	02-01-2014	12-02-2014
Cuarta Iteración	H17, H18, H19, H20	13-02-2014	20-03-2014	09-04-2014

3ª Fase: Iteraciones. (Metodología)

● VELOCIDAD DEL PROYECTO

Horario de Trabajo Semanal				
Lunes	Martes	Miércoles	Jueves	Viernes
8:00 am 12:00pm	8:00 am 12:00pm	8:00 am 12:00pm	8:00 am 12:00pm	8:00 am 12:00pm
Receso	Receso	receso	Receso	Receso
1:00pm5:00pm	1:00pm5:00pm	1:00pm5:00pm	1:00pm5:00pm	1:00pm5:00pm

	ITERACIÓN 1	ITERACIÓN 2	ITERACIÓN 3
Horas	90	60	56
Semanas	3	2	2
horas semanales	30	30	28
historia de usuario (velocidad del proyecto)	4	3	3

3ª Fase: Iteraciones. (Práctica)

● VELOCIDAD DEL PROYECTO

Horario de Trabajo Semanal				
Lunes	Martes	Miércoles	Jueves	Viernes
08:00pm	08:00pm	08:00pm	08:00pm	08:00pm
13:00pm	13:00pm	13:00pm	13:00pm	13:00pm

	ITERACIÓN 1	ITERACIÓN 2	ITERACIÓN 3	ITERACIÓN 4
Horas	200	135	250	100
Semanas	7	5,4	6	4
horas semanales	29	34	42	25
H .de usuario	7	4	5	4

3ª Fase: Iteraciones. (Metodología)

● REUNIONES DIARIAS

FECHA DEFINICIÓN DE LA HISTORIA			PROCESO	DESCRIPCIÓN TAREA	TIPO DE TAREA	RESPONSABLE	FECHA SEGUIMIENTO			ESTADO	OBSERVACIONES
DIA	MES	AÑO					DIA	MES	AÑO		
										CUMPLIDA/ PENDIENTE	
										CUMPLIDA/ PENDIENTE	
										CUMPLIDA/ PENDIENTE	

3ª Fase: Iteraciones. (Práctica)

REUNIONES DIARIAS

FECHA DEFINICIÓN DE LA HISTORIA			PROCESO	DESCRIPCIÓN TAREA	TIPO DE TAREA	RESPONSABLE	FECHA SEGUIMIENTO			ESTADO	OBSERVACIONES
DIA	MES	AÑO					DIA	MES	AÑO		
09	09	2013	Aspirantes	Diseño de Formulario con campos requeridos	Diseño	Javier Játiva Luis Égas	10	09	2013	CUMPLIDA	Ninguna
10	09	2013	Aspirantes	Validación de campos	Programación	Javier Játiva Luis Égas	11	09	2013	CUMPLIDA	Ninguna
11	09	2013	Aspirantes	Script para el ingreso de nuevo aspirante	Programación	Javier Játiva Luis Égas	12	09	2013	CUMPLIDA	Ninguna
12	09	2013	Aspirantes	Validación de ingreso	Programación	Javier Játiva Luis Égas	13	09	2013	CUMPLIDA	Ninguna
13	09	2013	Aspirantes	Pruebas de funcionamiento	Pruebas	Javier Játiva Luis Égas	16	09	2013	CUMPLIDA	Ninguna
16	09	2013	Instructores	Diseño de Formulario con campos requeridos	Diseño	Javier Játiva Luis Égas	17	09	2013	CUMPLIDA	Ninguna

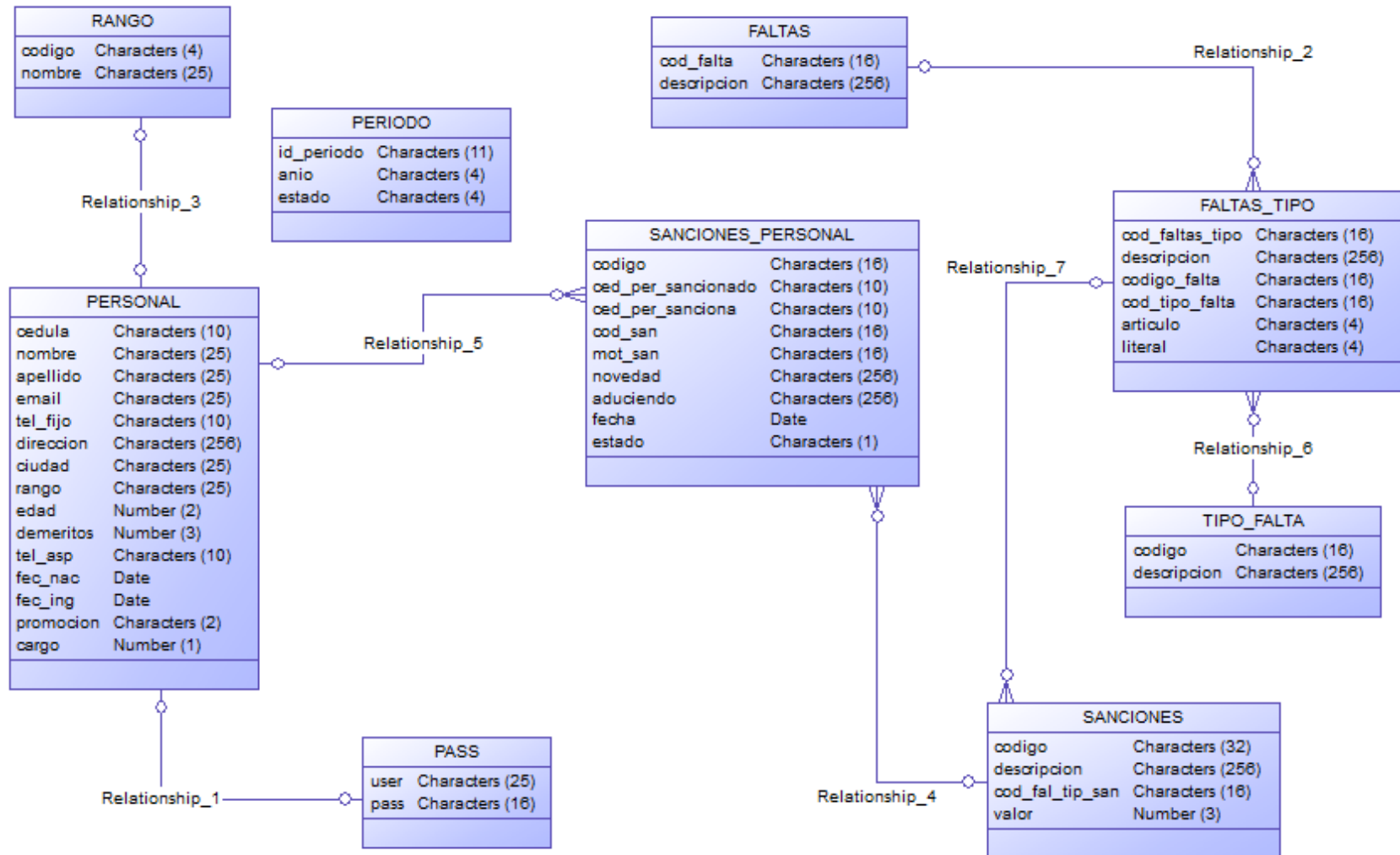
3ª Fase: Iteraciones. (Metodología)

- **MODELO ENTIDAD RELACIÓN**

Entidad:	Función:
Atributo	Descripción
...	

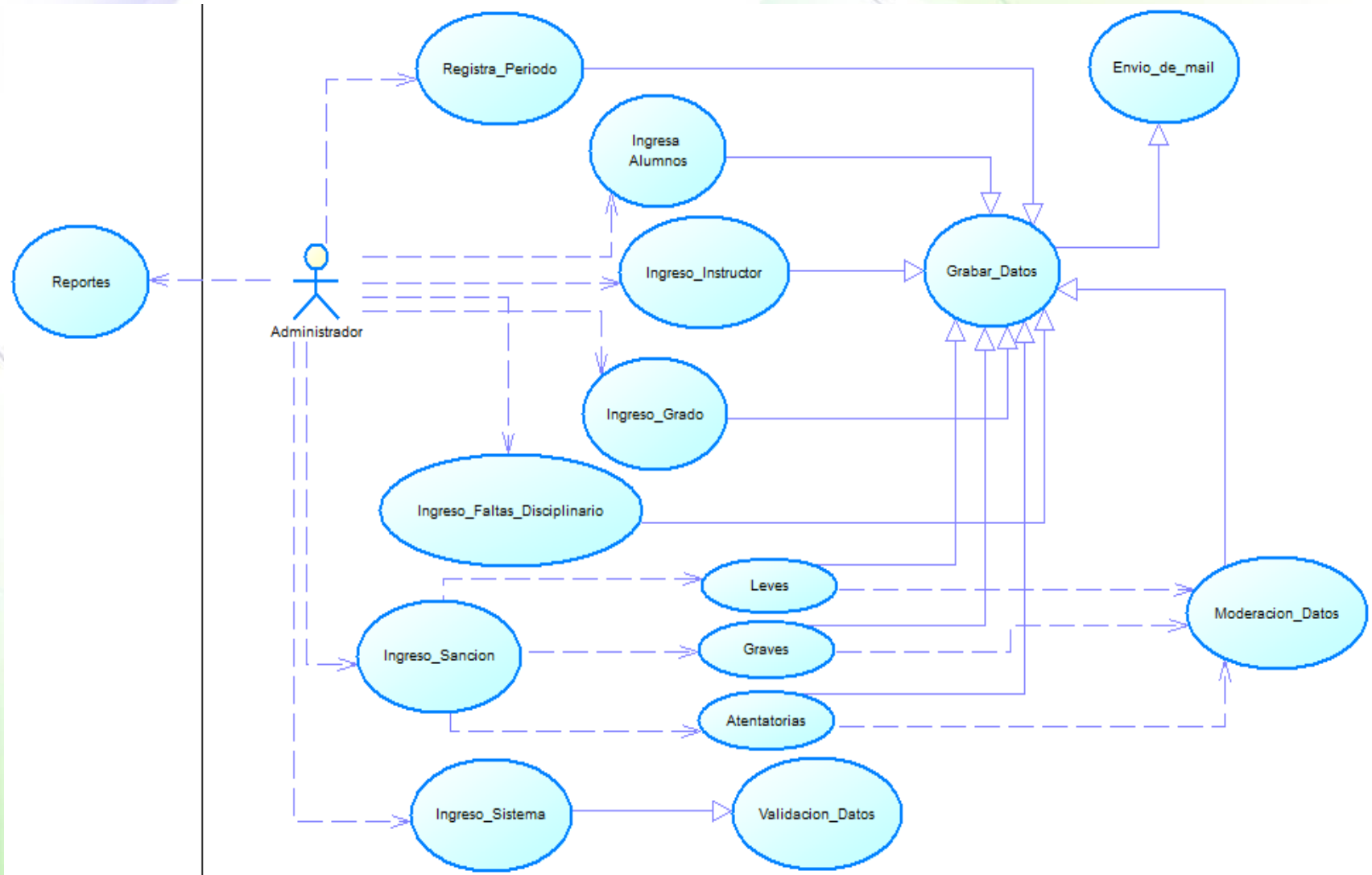
3ª Fase: Iteraciones. (Práctica)

MODELO ENTIDAD RELACIÓN



3ª Fase: Iteraciones. (Práctica)

● DIAGRAMA DE CASOS DE USO DE NEGOCIO



3ª Fase: Iteraciones. (Metodología)

● GLOSARIO DE TÉRMINOS

Término	Significado
TIC	Tecnologías de Información y Comunicación
Sistema COND_ATROS	Sistema Control de CONDUCTA de Aspirantes a TROpa de la ETFA
HU	Historia de Usuario
DFD	Diagrama de Flujo de Datos
Nómina	Listado en general de artículos

3ª Fase: Iteraciones. (Práctico)

● GLOSARIO DE TÉRMINOS

Término	Significado
TIC	Tecnologías de Información y Comunicación
Sistema COND_ATROS	Sistema Control de Conucta de Aspirantes a Tropa de la ETFA
HU	Historia de Usuario
DFD	Diagrama de Flujo de Datos
Nómina	Listado en general de artículos

3ª Fase: Iteraciones. (Metodología)

● DISEÑOS DE INTERFAZ



- Color fondo interfaz ingreso: #0866C6
- Texto encabezado: fuente Verdana, tamaño 34 puntos, color #000000
- Cajas de Texto: fuente texto Verdana, tamaño 14 puntos, color texto #999DC3, fondo cajas #FFFFFF, dimensión cajas 6 cm x 1cm.
- Botones de comando: fuente texto Verdana, tamaño 15 puntos, color texto #FFFFFF, color fondo #1E82E, dimensión botones 6 cm x 1cm
- Check Box: fuente texto Verdana, tamaño 10 puntos, color texto #FFFFFF.
- Logotipo: Imagen logotipo ETFA genérico para toda aplicación. Tamaño 10cm x 8cm.

3ª Fase: Iteraciones. (Práctico)

● DISEÑOS DE INTERFAZ

Sistema de Sanciones al P... x

etfa.nibble.com.ec/index.php

Escuela Técnica Fuerza Aérea

Ingreso Usuario

Ingreso Contraseña

INGRESAR

Mantener sesión

© 2013.

ES 21:17
12/05/2014

3ª Fase: Iteraciones. (Metodología)

● DISEÑOS DE INTERFAZ

The image shows a wireframe of a web interface with several key components and annotations:

- Header:** A blue navigation bar containing a **LOGO**, **USER**, and four menu items: **MENU1**, **MENU2**, **MENU3**, and **MENU4**. On the right side of the header, there is a **FOTO** placeholder and a user profile section with the text "usuario email@ejemplo.com" and links for **Perfil**, **Ajustes**, and **Salir**.
- Content Area:**
 - Mapa:** A section titled "MAPA1" with a sub-section "PRINCIPALES TAREAS" containing "Imagen" and "Ubicación".
 - Accesos Directos:** A section titled "ACCESOS DIRECTOS" with three blue buttons labeled "ACCESO 1", "ACCESO 2", and "ACCESO 3".
 - Tabla:** A table with three columns labeled "ITEM 1", "ITEM 2", and "ITEM 3".
 - Busqueda:** A search bar labeled "BUSQUEDA" with a magnifying glass icon.
 - Importante:** A yellow box titled "IMPORTANTE!" containing an "ANUNCIO" (announcement) with a "Ver..." button.
 - Leyenda:** A legend section at the bottom of the page.

Annotations (circled text with arrows pointing to specific elements):

- "LOGO ETFA O NOMBRE DEL DPTO." points to the LOGO.
- "MENU DE LOS PRINCIPALES PROCESOS" points to MENU1.
- "INFORMACION DE USUARIO EN LINEA" points to the user profile section.
- "ACCESOS DIRECTOS RELEVANTES" points to ACCESO 3.
- "ANUNCIO RELEVANTE O ESPACIO PARA PUBLICAR DOCUMENTOS" points to the ANUNCIO box.
- "SE PUEDE USAR PARA DESPLEGAR INFORMACION O TABLAS" points to the table.

3ª Fase: Iteraciones. (Práctico)

● DISEÑOS DE INTERFAZ

The screenshot shows a web browser window with the URL `etfa.nibble.com.ec/dashboard.php`. The page features a blue header with a logo, the word "Admin", and navigation icons for "Sancionados" (4), "Personal" (10), "Reportes" (10), and "Grado". A user profile for "xavier jativa" is visible in the top right.

The main content area is titled "Escritorio" and includes a search bar. Below this, there are three direct access buttons: "Sanción", "Listado", and "Manual".

An "IMPORTANTE!" alert box is present, stating: "Atención! Si ha olvidado ciertas normas de sanción se recomienda leer el PDF de las faltas." with a "Ver..." button.

At the bottom, there is a table header for "SEMANA ACTUAL" with columns: "SANCIONADO", "FALTA", "MOTIVO", and "DEMERITOS".

Footer text includes "© 2013. All Rights Reserved." and "Diseñado por: Egas-Játiva".

SANCIONADO	FALTA	MOTIVO	DEMERITOS
------------	-------	--------	-----------

3ª Fase: Iteraciones. (Metodología)

- **PRUEBAS**
- **Pruebas Unitarias**
- Estas pruebas se realizan para controlar el funcionamiento de un módulo (unidad de código) como puede ser subprogramas o métodos. Las pruebas unitarias aseguran que un determinado módulo cumpla con un comportamiento esperado en forma aislada antes de ser integrado al sistema.

3ª Fase: Iteraciones. (Práctico)

Caso Prueba Unitaria

Id Caso de Prueba: 01

Nombre Módulo: Gestión de Aspirantes

Iteración: 1-2-3

Descripción de la prueba: Alta, Modificación y baja de Aspirantes

Nombre del Tester: Luis Egas

Funcionalidad: Descripción	Método Utilizado	Recibe	Resultado esperado del método	Resultado esperado de la prueba	Resultado real de la prueba
Envío de información para el alta del Aspirante en la Base de datos	POST	Función de alta de aspirante programada en PHP	Return 1	Ok	Datos de aspirante ingresados correctamente
Envío de información para la modificación del Aspirante a la Base de datos	POST	Función de modificación de aspirante programada en PHP	Return 1	Ok	Datos de aspirante modificados correctamente
Envío de información para la baja del Aspirante a la Base de datos	POST	Función de baja de aspirante programada en PHP	Return 1	Ok	Aspirante inactivo.

3ª Fase: Iteraciones. (Metodología)

- **PRUEBAS**
- **Pruebas de aceptación**
- Las pruebas de aceptación, también llamadas pruebas funcionales, son pruebas de caja negra definidas por el cliente para cada historia de usuario, y tienen como objetivo asegurar que las funcionalidades del sistema cumplen con lo que se espera de ellas.

3ª Fase: Iteraciones. (Práctico)

Nombre: Alta de Aspirantes

Descripción: Ingreso de Aspirante.

Condiciones de Ejecución:

1. Todos los campos del Aspirante deben estar llenos, son obligatorios.
2. Los campos de texto (nombre, apellido, ciudad) se permite solo letras.
3. Los campos numéricos (cedula, teléfono, edad, fecha de nacimiento) se permite solo números.

Entrada / Pasos de ejecución:

1. Validación de campos.
2. Ingreso no duplicado de aspirante.
3. Ingreso de datos del aspirante.

Resultado Esperado: Datos ingresados correctamente.

Evaluación de la Prueba: Ok