



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



Departamento de Ciencias de la Computación

Carrera de Ingeniería en Software

Tema:

“Diseño de un modelo de trabajo basado en las metodologías tradicionales y no tradicionales implementando las mejores prácticas de la industria de desarrollo de software en la empresa de ACR DIGITAL S.A.S.”.

Autores:

**Llamuca Manzano, Andrea Lizbeth
Pérez Guanopatin, Lorena Mishel**

Director:

Ing. Espinosa Gallardo, Edison Gonzalo

Latacunga, 2024



**«Las grandes
oportunidades nacen de
haber sabido aprovechar
las pequeñas»**

Bill Gates



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Índice

01	Presentación del problema	<ul style="list-style-type: none">• Planteamiento y formulación del problema• Justificación e importancia• Objetivos• Hipótesis
02	Marco Teorico	<ul style="list-style-type: none">• Historia de la Fuerza Aérea Ecuatoriana• Sistema Web• Usabilidad• Scrum
03	Análisis, diseño y desarrollo del sistema web	<ul style="list-style-type: none">• Elicitación de requisitos de software• Diseño de arquitectura de software• Desarrollo
04	Análisis de resultados	<ul style="list-style-type: none">• Validación de software• Analisis estadistico• Resultados finales
05	Conclusiones y recomendaciones	<ul style="list-style-type: none">• Conclusiones• Recomendaciones



Resumen

En la actualidad para realizar un software primero debemos relacionar su desarrollo con las mejores prácticas, el cual, define modelos o diseños de referencia que proporcionan un plan o protocolo para optimizar las estructuras técnicas, organizativas y de proceso.

- El enfoque de este trabajo se basa en la factibilidad de aplicar metodologías y mejores prácticas de desarrollo de software en la empresa ACR DIGITALD S.A.S. El análisis exhaustivo de la literatura y la selección cuidadosa de artefactos y plantillas permitieron diseñar un modelo de trabajo eficiente y adaptado a las necesidades específicas de la empresa.



Presentación del problema

Planteamiento del problema

Crisis del software

Las metodologías de desarrollo de software han ido evolucionando desde los años 40, desde el principio de las computadoras, donde no se contaba con estándares, todo era artesanal, lo que terminaba en proyectos fallidos

Principal desafío

El principal desafío de la ingeniería de software es que se desarrollen los productos software con calidad

Actualidad

En la industria de software muchos de los sistemas no contienen una guía específica que trate acerca de las mejores prácticas en el proceso que se desarrolla el software, el cual, debe ajustarse a la necesidad del software y su estructura

Cambio en el paradigma

Las mejores prácticas están tomando en la industria de desarrollo de software un gran impacto, debido a que optimizan su desarrollo, dándole así también un valor agregado tanto en innovación como en proceso.

Formulación del problema

¿Cómo puede ACR DIGITAL S.A.S. integrar de manera efectiva las metodologías tradicionales y no tradicionales en su proceso de desarrollo de software para optimizar la eficiencia y calidad del producto final?

¿Cuáles son las mejores prácticas de la industria de desarrollo de software que pueden ser aplicadas específicamente a los procesos y proyectos de ACR DIGITAL S.A.S. para mejorar la productividad y la entrega de soluciones de alta calidad?

¿Cómo se puede gestionar el cambio organizacional y capacitar al equipo de ACR DIGITAL S.A.S. para asegurar una transición exitosa hacia el nuevo modelo de trabajo, incorporando las metodologías y prácticas seleccionadas?

Justificación e importancia



Búsqueda de calidad

- En la actualidad para realizar un software primero debemos relacionar su desarrollo con las mejores prácticas y recursos.



Mejores prácticas

- Las mejores prácticas están tomando en la industria de desarrollo de software un gran impacto



AGILE

- Creación de un modelo de Desarrollo ágil



Objetivo General

- Elaborar un enfoque laboral que integre tanto metodologías convencionales como alternativas con el fin de respaldar el progreso del desarrollo de software, incorporando las prácticas más destacadas de la industria del desarrollo de software en ACR DIGITALS S.A.S.

Objetivos Específicos

Construcción del marco teórico para fundamentar las metodologías y modelos tradicionales y no tradicionales de desarrollo de software.

Diseñar un modelo de trabajo para ayudar al proceso de desarrollo de software mediante la implementación de las mejores prácticas.

Implementar el modelo de trabajo que ayude y beneficie el desarrollo de software mediante la implementación de las mejores prácticas de la industria de desarrollo de software.



Hipótesis

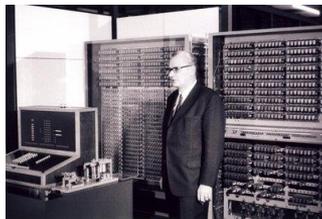
- Si se elabora un modelo de trabajo basado en las metodologías tradicionales y no tradicionales entonces se dispone de un modelo para soporte de desarrollo de software mediante la implementación de las mejores prácticas de la industria de desarrollo de software en la empresa ACR DIGITALS S.A.S.

Marco teórico

Primera etapa cronológica (1940-1970)



A partir de la década de los 40 nos encontramos en el inicio de la generación de las computadoras.



En la década de 1960, surge el modelo "Code and Fix" (codificar y corregir), que refleja la anarquía de los primeros años y marca el comienzo de la búsqueda de una base más sólida para la fabricación de software.



Los proyectos más comunes que fracasaban dando pérdidas hasta millonarias eran los sistemas de aeropuertos



En 1968, en Alemania, se acuña el término "Ingeniería de Software", definido como un "enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento del software"

Segunda etapa cronológica (1970-2000)



A partir de los años 70 comienzan a resolver los problemas complejos, lo que permite crear etapas para el desarrollo de software



En la segunda mitad de la década de 1970, surge el término "Ciclo de Vida del Software" (SDLC, por sus siglas en inglés, Software Development Life Cycle).



Entre 1970 y 1988 aparecen los "modelos tradicionales de desarrollo de software" y la diferencia entre modelos y metodologías.



En 1968, en Alemania, se acuña el término "Ingeniería de Software", definido como un "enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento del software"

Marco Teórico

Tercera etapa cronológica (2000-Presente)



Con la creciente demanda del software, lleva al crecimiento a métodos más simples y rápidos.



El Manifiesto Ágil marcó el inicio, siendo un documento que resume la filosofía "ágil"



Entre las metodologías ágiles más destacadas se encuentran: Scrum, Crystal Clear (cristal transparente), programación extrema (en inglés eXtreme Programming o XP), desarrollo de software adaptativo (ASD), feature driven development (FDD), Método de desarrollo de sistemas dinámicos.



A partir del año 2001 miembros de la comunidad de desarrollo de software en Utah, USA las proponen por 17 representantes de la industria del software como metodologías ágiles y así promueven el desarrollo ágil de aplicaciones

Marco Teórico

Scrum

Es un marco que permite la colaboración entre equipos y anima a los equipos a aprender a través de las experiencias

Product Backlog

listado de todas las tareas que queremos realizar durante el desarrollo del proyecto

Equipo Scrum

Dueño del producto
Equipo de desarrollo
Scrum Master

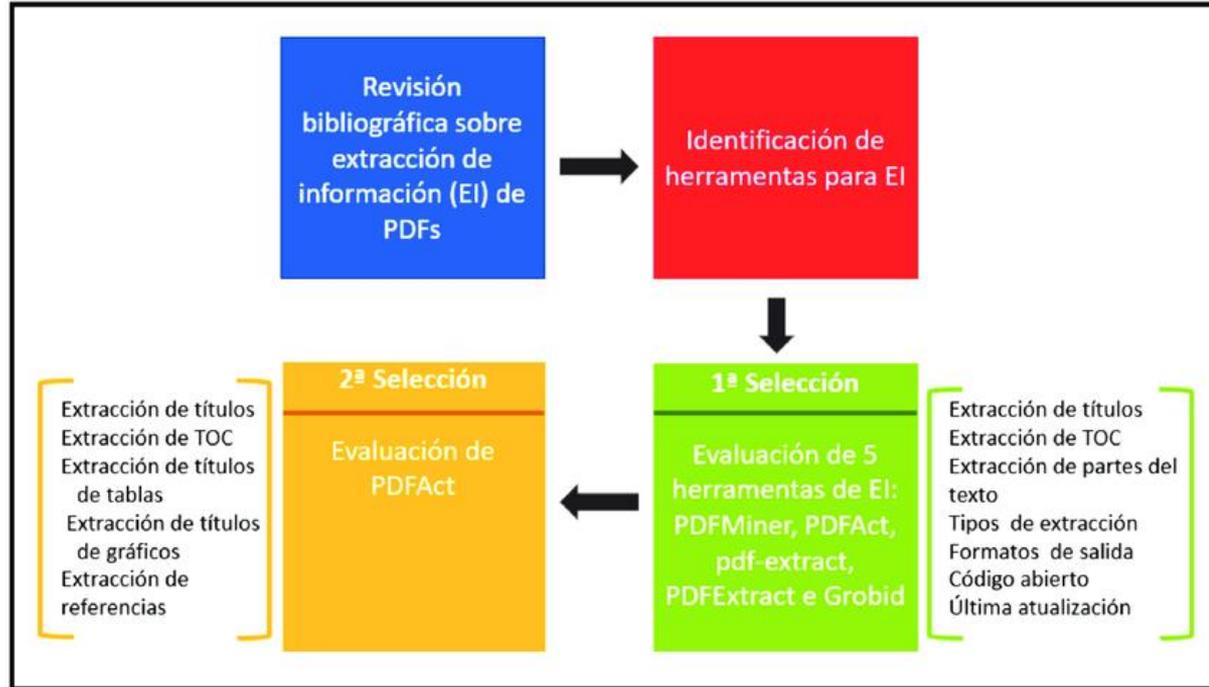
Eventos

hay eventos preestablecidos para crear regularidad y reducir la necesidad de reuniones no programadas en Scrum

Análisis, diseño y desarrollo

Fases

Etapas



Diseño de arquitectura de software

Código fuente

- Vistas del modelo 4+1 que se utilizan para detallar la arquitectura de software del presente proyecto de investigación, a través del uso de diagramas que están basado en el lenguaje de modelamiento unificado (UML).



Configuración de las herramientas

Productos



a) IntelliJ IDEA



b) Visual Studio Code

PostgreSQL



c) PostgreSQL

Entorno de trabajo



spring

a) Java Spring



VUE JS

b) Vue.js

Instalación de entornos y software



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Creación del proyecto base

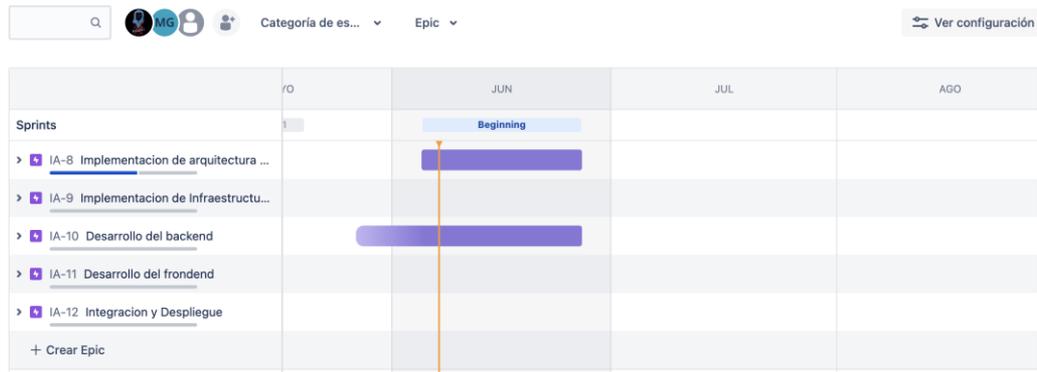
The screenshot shows a web browser window with the following details:

- Browser tabs: WhatsApp, El Sabor de la Tía, RESTAURANTE: Lucidchart
- Address bar: 127.0.0.1:8000/usuario/create
- Page title: El Sabor de La Tía
- Navigation menu: EMPRESA, CLIENTE, ALMUERZO, MENU, REPORTES, Contact, ANDREA LLAMUCA
- Section title: Crear Clientes
- Instruction: Los campos con * son obligatorios
- Form fields:
 - Usuario* (input type="text" value="Nombre del Usuario")
 - Contraseña* (input type="password" value="Contraseña del Usuario")
 - Repetir Contraseña* (input type="password" value="Contraseña del Usuario")
 - Empleado (dropdown menu with "SELECCIONE" selected)
 - Rol (dropdown menu with "SELECCIONE" selected)
- Buttons: Crear (blue), Cancelar (red)

Implementación

Hoja de ruta

[Dar feedback](#) [Compartir](#) [Exportar](#) [...](#)

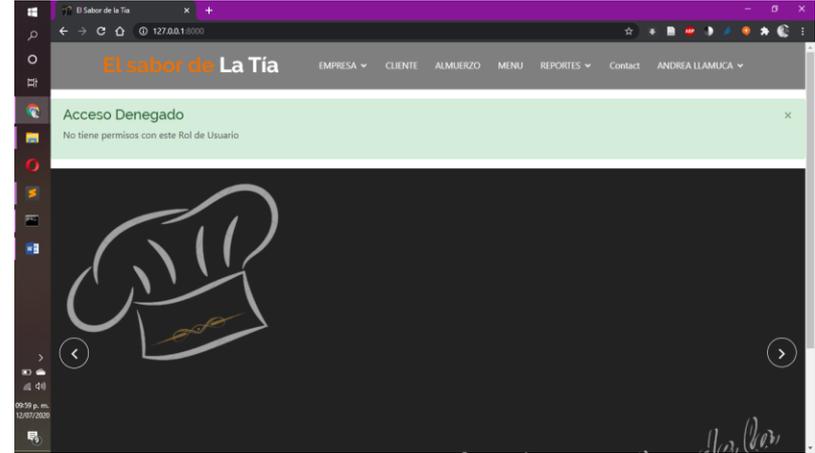
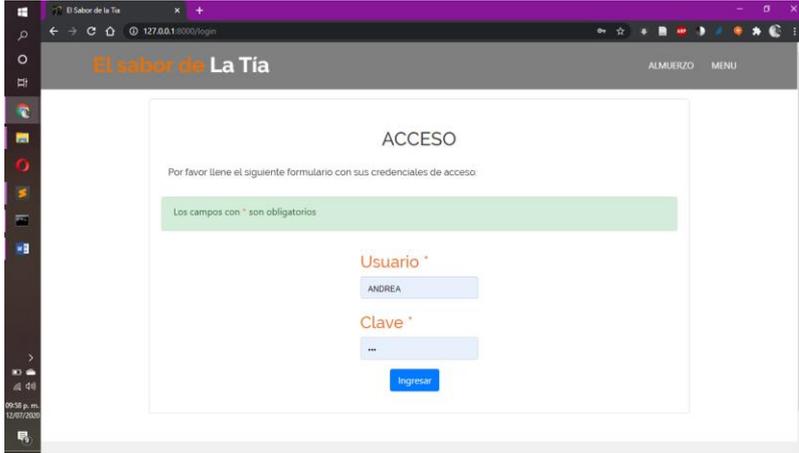


Implementación
de código Scrum

**cronograma de
actividades a cumplirse en
periodos de 4 semanas**

Análisis de resultados

Validación del sistema



Conclusiones y recomendaciones

Conclusiones

- El desarrollo de esta tesis, basado en el análisis bibliográfico académico permitió obtener un modelo para soporte de desarrollo de software, mediante la implementación de las mejores prácticas, en la industria de desarrollo de software para la empresa ACR DIGITALS S.A.S.
- La integración de diferentes metodologías tanto tradicionales y no tradicionales puede proporcionar un enfoque más completo y adaptable para afrontar las necesidades específicas en cada proyecto. Los equipos de desarrollo pueden enfrentar los desafíos cambiantes del proyecto y entregar un producto final de alta calidad.
- Con base en la investigación realizada, es práctico describir cada fase en el desarrollo de software, se ha podido identificar una serie de técnicas comunes y esenciales en el proceso. En la etapa de planificación se destacan elementos como las historias de usuario, los casos de uso, los diagramas de clases y de secuencia, así como el modelo conceptual.
- Los estudios revisados de diferentes técnicas y procedimientos comunes obtenidos de desarrollo de software, son fundamentales para mejorar el mismo. Estas técnicas, ofrecen un conjunto de prácticas para optimizar el proceso de desarrollo de software en diversas áreas.
- La comunicación efectiva, la colaboración, la formación adecuada y las revisiones periódicas son elementos clave para el éxito de la aplicación de las mejores prácticas de las metodologías tradicionales y no tradicionales.
- Los estudios revisados ofrecen valiosas técnicas en pruebas funcionales, pruebas de aceptación y pruebas unitarias durante la fase de pruebas de software, las cuales desempeñan una función fundamental en la obtención de productos de alta calidad. Estas técnicas, optimizan el proceso de desarrollo de software y garantizar la detección temprana de errores y la validación efectiva de la funcionalidad del software, lo que contribuye significativamente a la mejora continua y al cumplimiento de los estándares de calidad en cada etapa del ciclo de desarrollo.
- El modelo desarrollado no solo se alinea con los principios del método ágil clásico, sino que también mejora la disciplina del proceso al incorporar opciones de gestión. Esto facilita la creación de documentos y artefactos, apoyando a los desarrolladores a lo largo del proceso de desarrollo de software.

Recomendaciones

- Analizar cuidadosamente las necesidades y características del proyecto antes de seleccionar una metodología. No existe un enfoque único que sea adecuado para todos los proyectos.
- Considerar la posibilidad de utilizar un enfoque híbrido que combine elementos de diferentes metodologías. Esto permite adaptar el enfoque a las necesidades específicas del proyecto y aprovechar lo mejor de ambos enfoques.
- Fomentar la comunicación y la colaboración efectivas dentro del equipo. Esto es esencial para el éxito de cualquier metodología, ya que facilita la coordinación, el intercambio de ideas y la resolución de problemas.
- Proporcionar una formación adecuada a los miembros del equipo sobre la metodología seleccionada. Esto asegura que todos estén alineados y tengan una comprensión clara de los principios y prácticas a seguir.
- Realizar revisiones periódicas del progreso del proyecto y la eficacia de la metodología.

Gracias por su atención