



Universidad Nacional de Ingeniería
Facultad de Electrotecnia y computación

**TRABAJO MONOGRÁFICO PARA OPTAR AL TÍTULO
DE INGENIERO EN COMPUTACIÓN**

**Sistema de Manejador de Contenido Web para el Programa de
Investigación en TIC (UNI-Nicaragua)**

Presentado Por

Br. Alvaro Javier Romero Parrales

Br. Sergio Israel Polanco Pao

Tutor

TKL. Johnny Flores Delgadillo

Managua, agosto 2018

**“Sistema de Manejador de Contenido Web para Programa de
Investigación en TIC (UNI-Nicaragua)”**

Agradecimientos:

Agradecemos hoy y siempre a nuestras familias por el esfuerzo realizado, el apoyo incondicional durante nuestra formación personal y profesional y por otorgarnos la fortaleza necesaria para seguir adelante.

De igual forma agradecemos a todos los maestros que nos enseñaron a valorar nuestros estudios y nos ayudaron a superarnos cada día.

Agradecimientos de manera muy especial a nuestro tutor Johnny Flores Delgadillo por guiarnos de inicio a fin en este reto complejo lleno de obstáculos.

Nuestros más sinceros agradecimientos a cada uno de ustedes.

Resumen:

En el pasado las herramientas para la elaboración de sistemas web eran básicamente editores que permitían generar un documento web y que en el transcurso del tiempo evolucionaron para incorporar el control de la estructura y otras funcionalidades, pero en general estaban enfocadas más a la creación que al mantenimiento. En los últimos años se ha desarrollado el concepto de Sistema manejador de contenido (CMS), que son un conjunto de herramientas que se desarrollan con el objetivo de permitir crear y mantener un sistema web con facilidad, encargándose de los trabajos más tediosos que hasta ahora ocupaban el tiempo de los administradores de los sitios web.

El programa de investigación en TIC de la Universidad Nacional de Ingeniería (UNI) en Nicaragua, requiere constantemente agregar y actualizar información, así como también evidenciar los resultados de sus propias investigaciones y proyectos en colaboración, incluyendo programas académicos a nivel de posgrado.

Esta necesidad motiva la creación de un Sistema manejador de contenido web (CMS) que aporte herramientas para definir la estructura, el formato de los documentos web, el aspecto visual, uso de patrones y un sistema modular que permita incluir funciones no previstas inicialmente, con el fin de aumentar el nivel de implementación del sistema de gestión integrado, en el marco del mejoramiento continuo de este.

Del resultado del análisis de los requerimientos se desarrollará el Sistema manejador de contenido (CMS) (ver definición en capítulo 2) mediante herramientas de software libre, que sea fácilmente escalable y que pueda estructurarse en diferentes vistas según el contexto. Las soluciones completas para la Gestión de Contenido Web (CMS) son complejas y no precisamente económicas. Por esa razón, resulta obligatorio reflexionar y concretar cuáles son las necesidades reales de la organización respecto a la gestión de su contenido.

Al final el lector tendrá una clara idea de la arquitectura y aspectos tecnológicos detrás de este tipo de sistemas que hoy en día son de gran uso al momento de implementar un sistema web.

Tabla de contenido

Introducción:	8
Objetivos	10
Objetivo General	10
Objetivos Específicos	10
Justificación:	11
METODOLOGÍA DE LA INVESTIGACIÓN	12
Metodología:	12
Diseño de investigación:	12
Distribución del contenido:	13
Capítulo 1: Aplicaciones Web	15
1.1 ¿Qué es un sitio web?	15
1.2 Arquitecturas de Aplicaciones Web	16
Capítulo 2: Sistema Manejador de Contenido Web (CMS)	18
2.1 Definición de CMS	18
2.2 Características básicas de un CMS	18
2.2 Funcionamiento de un CMS	20
2.3 Arquitectura de un CMS	21
2.4 Beneficios de un CMS	21
2.5 Clasificación de los CMS	22
Capítulo 3: Tecnología de desarrollo	24
3.1 Python	24
3.1.1 Características del lenguaje	24
3.1.1.1 Propósito general	24
3.1.1.2 Multiplataforma	24
3.1.1.3 Interpretado	25
3.1.1.4 Interactivo	25
3.1.1.5 Orientado a Objetos	25
3.1.1.6 Funciones y librerías	25

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

3.1.1.7 Sintaxis clara	25
3.2 JavaScript	25
3.2.1 Características	26
3.2.1.1 Imperativo y estructurado	26
3.2.1.2 Dinámico	26
3.2.1.3 Orientado a Objetos	27
3.2.1.4 Evaluación en tiempo de ejecución	27
3.2.1.5 Funcional	27
3.3 FrameWorks	27
3.4 Django	28
3.5 jQuery	28
3.6 Bootstrap	29
3.6.1 Ventajas de usar Bootstrap	29
Capítulo 4: Metodologías de desarrollo ágil	30
4.1 Metodología SCRUM	30
4.1.1 Actividades a realizar	31
4.1.1.1 Sprint Planning	31
4.1.1.2 Sprint	32
4.1.1.3 Scrum Daily Meeting	32
4.1.1.4 Sprint Review	32
4.1.1.5 Sprint Retrospective	32
4.1.1.6 Los roles del equipo	32
4.1.1.7 Product BackLog	33
4.1.1.8 ScrumTaskboard	33
4.1.1.9 Burndown Chart	34
4.2 Metodología Extreme Programming (XP) o Programación Extrema	34
4.2.1 Ventajas y desventajas de la metodología XP	34
4.2.1.1 Ventajas	35
4.2.1.2 Desventajas	35
4.2.1.3 Beneficios	35

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

4.2.2	Ciclo de vida de la metodología XP	35
Capítulo 5: Elección de Metodología		36
Capítulo 6: Elección de Tecnologías de Desarrollo		38
Capítulo 7: Información SAREC		40
7.1	Historia	40
7.2	Organización	41
7.3	Líneas de Investigación	42
7.3.1	Sistemas y tecnologías de información	43
7.3.2	Sistemas de Radiocomunicaciones.	43
7.3.3	Telemedicina (aplicaciones médicas de procesamiento de señales)	44
7.3.4	Programa de Maestría en Gestión TIC	44
7.4	Publicaciones	45
7.5	Problemas identificados relacionados con el manejo de la información de PROTIC	45
7.6	Analizando la información recopilada	45
Capítulo 8: Sistema Manejador de Contenido Web para el Programa de Investigación en TIC		47
8.1	Reunión de planificación de Sprint	47
8.2	El Scrum Diario	47
8.3	Trabajo de desarrollo durante el Sprint	48
8.4	Revisión del Sprint	48
8.5	Retrospectiva del Sprint	49
8.6	Roles del proyecto	49
8.7	Requerimientos del sistema	50
8.8	Historias de usuario	51
8.9	Lista de historias de usuario por orden de prioridad (BACKLOG)	57
8.10	Definición de Sprints	58
8.11	Planificación de los Sprints	61
8.12	Desarrollo del Sistema	63
8.12.1	Sprint Numero 1	63
8.12.2	Sprint Numero 2	70
8.12.3	Sprint Numero 3	77

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

8.12.4 Sprint Numero 4	80
8.13 Liberación del Sistema	87
Capítulo 9: Conclusiones	88
Capítulo 10: Bibliografía	89

Introducción:

Hoy en día el mercado está más globalizado y la competencia es cada vez más fuerte, mostrar una imagen corporativa, los servicios y/o laborales desempeñadas está relacionado de manera positiva con el incremento de las ventas, la productividad y el valor de mercado de la organización, esto nos plantea un escenario en el que los usuarios saben lo que quieren y dónde encontrarlo y dar respuestas a sus necesidades es un desafío cada vez mayor en donde entran en juego los sitios web.

Es muy importante que la organización que tiene un sitio web cuente con un sistema que le permita agregar y actualizar los contenidos con frecuencia y libertad. No es un secreto que para obtener buen posicionamiento en buscadores es muy importante contar con contenido de calidad y fresca.

Como reflejo de tal afirmación, el programa de investigación en TIC de la Universidad Nacional de Ingeniería (UNI) en Nicaragua, requiere constantemente agregar y actualizar su información, así como también evidenciar los resultados de sus propias investigaciones y proyectos en colaboración, incluyendo programas académicos a nivel de posgrado.

Siendo consciente de tal necesidad se desarrollará un Sistema manejador de contenido web (CMS) (ver definición en capítulo 2) mediante herramientas de software libre, que sea fácilmente escalable y que pueda estructurarse en diferentes vistas según el contexto.

El trabajo consistió en el estudio de los procesos de creación y modificación del contenido publicado por los involucrados para lograr un entendimiento en profundidad de estos y poder modelarlos, determinando a partir de los modelos realizados las necesidades funcionales que se le exigirán al sistema a desarrollar.

La necesidad de implementar un sistema manejador de contenido fue soportada ya que dicho departamento posee distintos documentos web en los cuales se narra o reflejan distintas tareas o trabajos realizados, pero estos no están integrados como un sistema de información, poseen muy poca interrelación y no permiten administrar el contenido publicado por los integrantes de este. Ante la inexistencia de un sitio web organizado, se propone el desarrollo de un sitio web orientado hacia su visión, que permita publicar y administrar contenido referente a las actividades del departamento.

Este sitio web se vincula con todos los investigadores y miembros del departamento, el cual podrán administrar su propio contenido, por lo cual, se elabora un CMS que posee los siguientes beneficios: publicación de contenido, flexibilidad y escalabilidad, total independencia del diseño, entre otros.

Para elegir la plataforma y herramientas de desarrollo se realiza un diagnóstico del sitio web actual y se eligen dichos componentes basados en los resultados obtenidos y en los requerimientos establecidos.

Dichos componentes son elegidos debido a la facilidad y flexibilidad que aportan en el desarrollo de sitios web pensados a pequeña y gran escala y que nos permitirán realizar el sitio web de alta calidad.

Objetivos

Objetivo General

- Desarrollar un sistema manejador de contenido web para el Programa de Investigación en TIC (PROITIC) de la Facultad de Electrotecnia y Computación de la Universidad Nacional de Ingeniería.

Objetivos Específicos

- Analizar los requerimientos del sistema a partir de las necesidades transmitidas por los integrantes del Programa de Investigación en TIC.
- Diseñar el sistema manejador de contenido web para el Programa de Investigación en TIC mediante el uso de metodología SCRUM.
- Codificar el Sistema Manejador de Contenido Web para el programa de investigación en TIC en base a los requerimientos obtenidos utilizando el lenguaje de programación Python.
- Implantar el Sistema manejador de Contenido Web en el ambiente de producción.
- Realizar pruebas de funcionalidades y correspondencias de requerimientos establecidos.

Justificación:

El Programa de Investigación en TIC de la Universidad Nacional de Ingeniería (Nicaragua) ha crecido significativamente en información, se ha generado una gran cantidad de artículos científicos en diferentes conferencias internacionales en los últimos años, así mismo oferta programas de posgrados tal es el caso de MGTIC y la Especialidad en Gestión Estratégica de auditoría de TIC.

Esto ha dificultado la gestión de su información en una web estática, demandan una web dinámica que crezca a medida que crece la información del programa. Para ello es vital la creación de un Sistema Manejador de Contenido Web (CMS) (ver definición en capítulo 2) que responda a sus necesidades para creación o actualización del contenido. Asimismo, cada miembro del programa será responsable de agregar y actualizar información que le competa.

METODOLOGÍA DE LA INVESTIGACIÓN

Metodología:

A continuación, se presentará un marco de trabajo basado en las prácticas de SCRUM, aplicado para gestionar el diseño del Sistema de Manejador de Contenido Web para Programa de Investigación TIC. En la propuesta se aplica SCRUM desde la concepción de la idea, en el proceso de elaboración del proyecto y finalizando con su presentación final.

El Scrum es un proceso de la Metodología ágil que se usa para minimizar los riesgos durante la realización de un proyecto, pero de manera colaborativa.

Entre las ventajas se encuentran la productividad, calidad y el seguimiento diario de los avances del proyecto; logrando que los integrantes estén unidos, comunicados y que el cliente esté al corriente del progreso.

Diseño de investigación:

Inicialmente coordinaremos con el equipo de PROITIC para identificar sus necesidades y expectativas. Evaluaremos el sitio web actual, analizando el grado en que este cumple las necesidades del programa. Esto se concreta al definir Product Backlog, que contiene todos los requerimientos funcionales y no funcionales que deberá satisfacer los sistemas a construir. De esta manera estaremos logrando el objetivo específico número 1.

En segundo lugar, planificaremos el desarrollo del nuevo sistema, desde la Interfaz y experiencia de usuario, hasta la arquitectura del sistema. Así mismo, evaluaremos las herramientas de las que disponemos para el cometido (lenguajes, frameworks, etc). Logrando así el cumplimiento del objetivo específico número 2.

A continuación, procederemos con el desarrollo en cuestión. Trabajando en la lógica y el diseño del front-end y back-end; además construiremos las estructuras de almacenamiento necesarias (alcance del objetivo específico número 3).

Los dos pasos anteriores se desarrollarán a lo largo de una serie de iteraciones semanales llamadas sprints. En los sprints son planeadas las actividades para cada iteración, y que se constituyen en estimados, que dan la temática a los encuentros diarios de Scrum.

En el siguiente paso nos aseguraremos de que el sistema cumple los requisitos establecidos, invitando al equipo de PROITIC a usarlo en su fase beta.

Desarrollaremos también, sistemas de pruebas unitarias, funcionales, de integración y de estrés, para asegurar el correcto funcionamiento del sistema (objetivo específico número 5).

Finalmente desplegaremos el sistema en el ambiente de producción, sobre la infraestructura de la Universidad Nacional de Ingeniería, monitoreando su correcto funcionamiento (objetivo específico número 4).

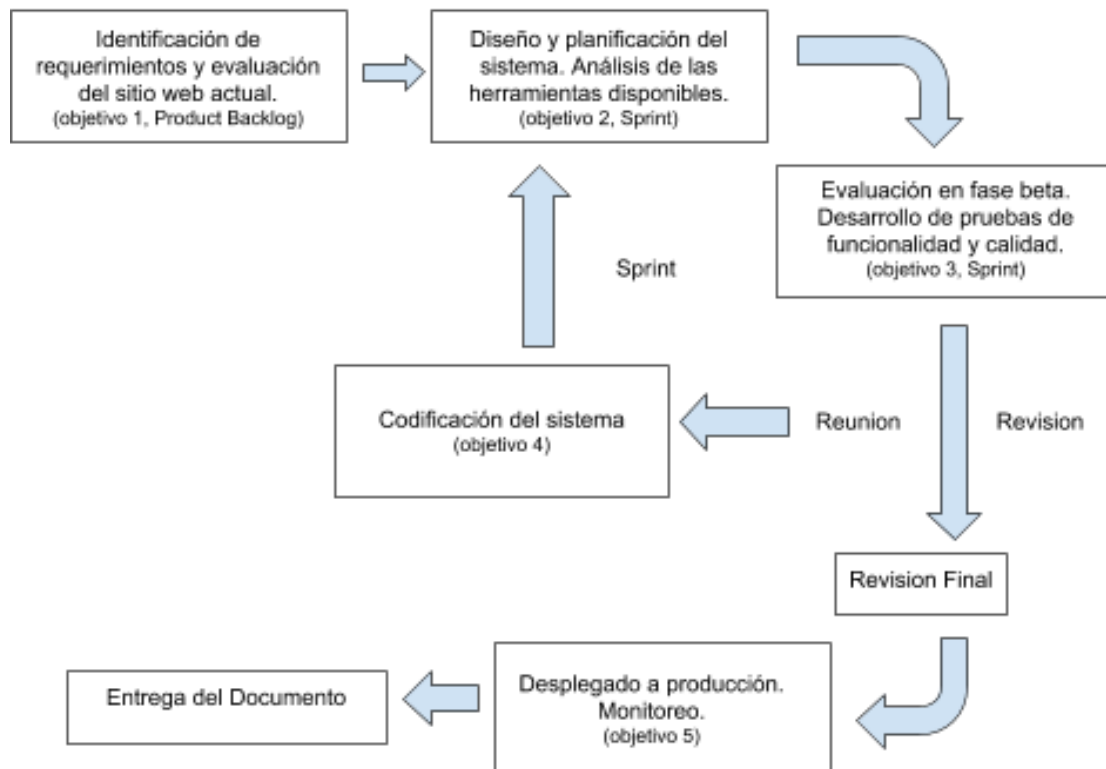


Figura N° 1. Diseño de Investigación

Distribución del contenido:

Para realizar la presente monografía y lograr la satisfacción de los objetivos planteados se desarrolló la siguiente metodología de trabajo:

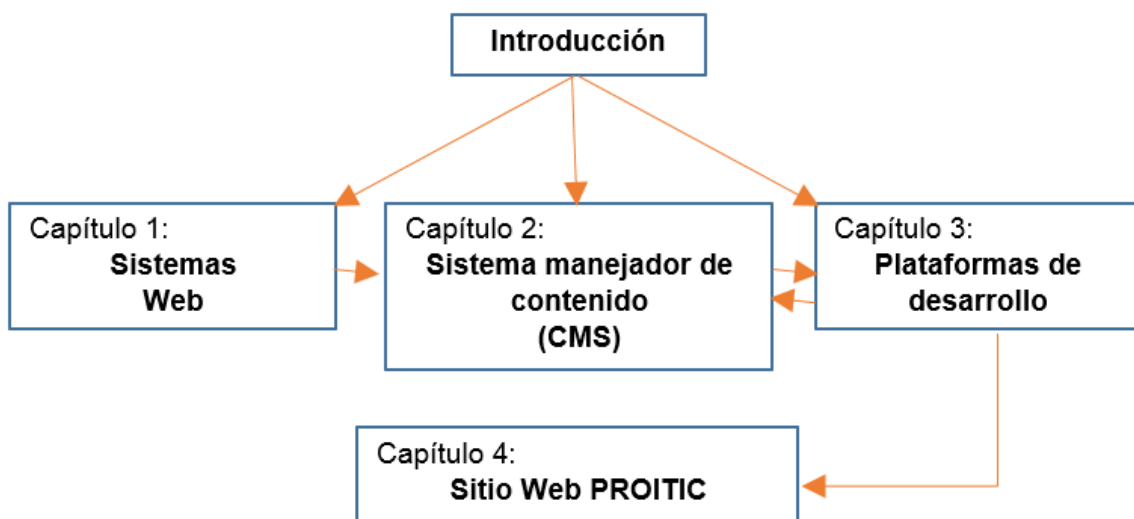


Figura N° 2. Diagrama de Contenido

En la sección introductoria se define el problema, la solución y aporte de la monografía. Desde la introducción podremos leer cualquier capítulo del documento para poder entender el resultado de capítulo 4.

Descripción corta de cada capítulo:

- ✓ Capítulo 1 - Sistemas Web: Abarca la introducción a los sitios web, su definición, historia y evolución.
- ✓ Capítulo 2 – CMS: Se define al Sistema manejador de contenido web (CMS), su estructura, su clasificación y se enumeran las ventajas y desventajas de los mencionados.
- ✓ Capítulo 3 – Plataformas de desarrollo: Se enumera y describe las plataformas y tecnologías a utilizar para el desarrollo e implementación del Sistema manejador de contenido web (CMS), lenguajes de programación, FrameWorks, etc.
- ✓ Capítulo 4 – Sitio Web PROITIC: Aborda al análisis del actual sitio Web PROITIC, así como el desarrollo del Sistema manejador de contenido (CMS) propuesto, presentando resultados de la implementación del mismo.

Capítulo 1: Aplicaciones Web

En un principio los sitios web eran simplemente texto plano, pero con el avance de la tecnología y la demanda en crecimiento de estos mismos, se generaron nuevas maneras de construirlos y diseñarlos. A medida que el tiempo avanzó estos sitios web fueron tomando nuevos aspectos, animaciones, funcionalidades. Ahora estos requerían ser vistosos, cómodos y fáciles de usar para el usuario final.

1.1 ¿Qué es un sitio web?

Las aplicaciones web son un tipo de software que se codifica en un lenguaje soportado por los navegadores web y cuya ejecución es llevada a cabo por el navegador en Internet o de una intranet [10]. Esto significa que los datos o los archivos en los que trabajas son procesados y almacenados dentro de una red a través de un navegador.

Las aplicaciones web han tomado mucha popularidad debido a la practicidad que ofrecen los navegadores web como clientes ligeros, la independencia del sistema operativo que uses en tu computador o dispositivo móvil, la facilidad para actualizar y mantener aplicaciones web sin la necesidad de tener que distribuir el software o que se tengan que instalar el mismo por los usuarios y el libre acceso de los usuarios en cualquier momento, lugar o dispositivo, sólo con tener conexión a Internet y los datos de acceso.

En las aplicaciones web suelen distinguirse tres niveles (como en las arquitecturas cliente/servidor de tres niveles): el nivel superior que interacciona con el usuario (el cliente web, normalmente un navegador), el nivel inferior que proporciona los datos (la base de datos) y el nivel intermedio que procesa los datos (el servidor web) [10].

El protocolo HTTP forma parte de la familia de protocolos de comunicaciones TCP/IP, que son los empleados en Internet. Estos protocolos permiten la conexión de sistemas heterogéneos, lo que facilita el intercambio de información entre distintos ordenadores.

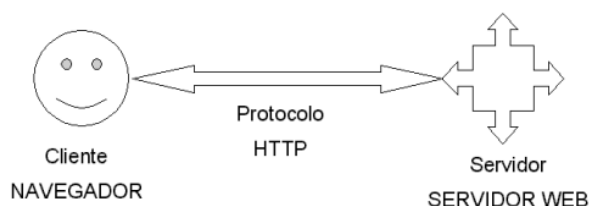


Figura N° 3. Protocolo HTTP

- ✓ **El Cliente:** El cliente web es un programa con el que interacciona el usuario para solicitar a un servidor web el envío de los recursos que desea obtener mediante HTTP [10]. La parte cliente de las aplicaciones web suele estar formada por el código HTML que forma la página web más algo de código ejecutable realizado en lenguaje de script del navegador (JavaScript). Por tanto, la misión del cliente web es interpretar las páginas HTML y los diferentes recursos que contienen (imágenes, sonidos, etc.).
- ✓ **El Servidor:** El servidor web es un programa que está esperando permanentemente las solicitudes de conexión mediante el protocolo HTTP por parte de los clientes web [10]. La parte servidor de las aplicaciones web está formada por páginas estáticas (documentos HTML) que siempre muestran el mismo contenido, recursos adicionales (multimedia, documentos adicionales, etc.) que se pueden emplear dentro de las páginas o estar disponibles para ser descargados y ejecutados (visualizados) en el cliente y programas o scripts que son ejecutados por el servidor web cuando el navegador del cliente solicita algunas páginas. La salida de este script suele ser una página HTML estándar que se envía al navegador del cliente.

1.2 Arquitecturas de Aplicaciones Web

Las aplicaciones web se basan en una arquitectura cliente/servidor: por un lado, está el cliente (el navegador, explorador o visualizador) y por otro lado el servidor (el servidor web). Existen diversas variantes de la arquitectura básica según como

se implementen las diferentes funcionalidades de la parte servidor. Las arquitecturas más comunes son:

1. Todo en un servidor: un único ordenador aloja el servicio de HTTP, la lógica de negocio y la lógica de datos y los datos. El software que ofrece el servicio de HTTP gestiona también la lógica de negocio (Figura N° 4).

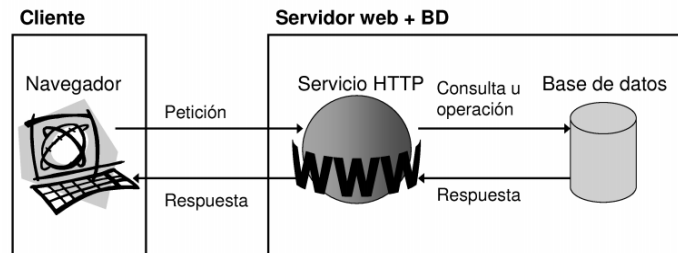


Figura N° 4. Arquitectura de sitio web basico

2. Servidor de datos separado: a partir de la arquitectura anterior (Figura N° 4), se separa la lógica de datos y los datos a un servidor de bases de datos específico.
3. Todo en un servidor, con servicio de aplicaciones: en la arquitectura número 1 se separa la lógica de negocio del servicio de HTTP y se incluye el servicio de aplicaciones para gestionar los procesos que implementan la lógica de negocio (Figura N° 5).

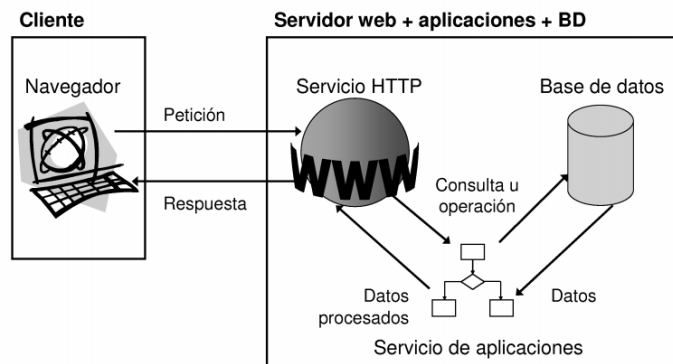


Figura N° 5. Arquitectura de sitio web con servidor de aplicaciones

Capítulo 2: Sistema Manejador de Contenido Web (CMS)

Cuando queramos crear una página web complicada por ejemplo un blog o un servidor de noticias vamos a tener opciones prediseñadas, los programadores se dieron cuenta que el código se estaba repitiendo constantemente, entonces lo que hicieron fue paquetes de código que ejecutaban una serie de tareas que eran habituales para todo los involucrados en dichas labores, a la larga esos paquetes de código se fueron complicando hasta llamarse gestores de contenido o como todos los conocen CMS.

2.1 Definición de CMS

Anteriormente el mantenimiento de un sitio web no era una tarea muy sencilla, era necesario el manejo de código HTML para poder crear y actualizar páginas. Pero ahora, existen sistemas que brindan sencillas y muy útiles herramientas para crear, actualizar y mantener sitios web.

En los últimos años se ha desarrollado el concepto de un instrumento que actúe como un sistema de gestión de contenidos, CMS:

Un Sistema de gestión de contenidos (Content Management System, CMS) permite la creación y administración de contenidos principalmente en páginas web. Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio [1].

El sistema manejador de contenido permite manejar de manera independiente el contenido y el diseño, de esa manera es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo público.

2.2 Características básicas de un CMS

Entre las características más importantes de los sistemas de gestión de contenidos y que constituyen una evolución de los sistemas de información

tradicionales, está el hecho de que sean manejables por todos los tipos de usuarios, todos pueden subir información, manipularla, representarla, etcétera (Figura N° 6).



Figura N° 6. Representación de actores de los CMS

La definición de CMS hace necesario generar toda una clasificación con diversas familias, géneros y subgéneros, entre los cuales resaltan algunas características genéricas que cumpliría en mayor o menor medida a la herramienta que quiera ser denominada como un CMS, ayudando así a los usuarios no conocedores del tema de la programación y los detalles que en el fondo le dan funcionalidad al sitio, de tener que trabajar con estos, dedicándose solo al contenido y a los pequeños puntos en el diseño, estas características comunes entre los CMS se listan a continuación:

- ✓ Separación del contenido de la presentación del mismo, se crea una plantilla que se puede modificar con el tiempo si se desea, pero que no va a afectar en ningún caso a los contenidos ya creados.
- ✓ La gestión de contenido resulta sencilla, no se necesita de ninguna aplicación ni de conocimiento técnico, basta un navegador web y conocer la interfaz de administración.
- ✓ Edición WYSIWYG, se dispone de editores de texto tipo Word o similar con el objetivo de crear/editar el contenido de manera que al usuario le sea familiar el flujo de trabajo.
- ✓ Control de seguridad y de usuarios.
- ✓ Gestionar, ofrecer, y restringir el acceso a contenido y también la edición de contenido.

2.2 Funcionamiento de un CMS

Los CMS nos permiten crear un sitio web o blog de una manera en la que nos es más sencillo crear, clasificar, editar y publicar la información que estos contengan, generalmente trabajan contra una base de datos de modo que el editor simplemente actualiza la base de datos, incluyendo la nueva información y editando la existente [4].

El autor James Robertson propone una división de la funcionalidad de los sistemas de gestión de contenidos en tres categorías:

- ✓ Creación de contenido

Un CMS aporta herramientas para que los creadores sin conocimientos técnicos en páginas Web puedan concentrarse en el contenido. Lo más habitual es proporcionar un editor de texto WYSIWYG, en el que el usuario ve el resultado final mientras escribe. Un CMS puede incorporar una o varias de estas herramientas, pero siempre tendría que proporcionar un editor WYSIWYG por su facilidad de uso.

- ✓ Gestión de contenido

Los documentos creados se depositan en una base de datos central donde también se guardan el resto de datos de la Web, cómo son los datos relativos a los documentos (versiones hechas, autor, fecha de publicación y caducidad, etc.), datos y preferencias de los usuarios, la estructura de la Web, etc.

- ✓ Publicación y presentación

Una página aprobada se publica automáticamente, y cuando caduca se archiva para futuras referencias. En su publicación se aplica el patrón definido para toda la Web o para la sección concreta donde está situada, de forma que el resultado final es un sitio Web con un aspecto consistente en todas sus páginas. Esta separación entre contenido y forma permite que se pueda modificar el aspecto visual de un sitio Web sin afectar a los documentos ya creados y libera a los autores de preocuparse por el diseño final de sus páginas.

2.3 Arquitectura de un CMS

A nivel operativo lo fundamental de un CMS es que dispone de dos partes [1]. Cuando vemos la página web en un navegador no vemos estas dos capas diferenciadas, no vemos la base de datos, ni la programación, ni el código fuente del diseño, lo que vemos es la suma de todas ellas, el resultado final. Es lo que se llama la renderización de la página web.

- ✓ **Sección web pública:** Es la página web como tal que ven las personas que acceden a ella.
- ✓ **Sección web privada:** Desde cualquier lugar con conexión a internet y un navegador con un usuario y contraseña podemos acceder a la parte interna “de trabajo” del gestor de contenidos. Allí según el perfil que tengamos (es decir quien seamos y que permisos tenemos) mediante páginas web de uso interno llamadas “maquetadores” podemos actualizar el contenido de la página web pública, modificarlo, corregirlo y crear contenido nuevo.

Un CMS a nivel de su estructura se distingue en 3 capas [1].

- ✓ **Base de datos:** La base de datos tiene guardado todo el contenido que se ha escrito en la web, así como muchos de los parámetros de configuración, categorías, organización, usuarios y contraseñas.
- ✓ **Programación:** Está contenida en los ficheros de la web. Lo que hacen estos ficheros al ejecutarse es solicitar la información que el usuario ha pedido desde el navegador de internet y extraerla para mostrarla al usuario de forma ordenada y estructurada “insertándose” en los lugares que le corresponde dentro del diseño de la página web.
- ✓ **Diseño:** Define el diseño de la web, es decir “la maqueta” dibujando y estilizando el contenido que la programación se encarga de extraer de la base de datos.

2.4 Beneficios de un CMS

De acuerdo a sus funcionalidades un sistema gestor de contenidos proporciona los siguientes beneficios:

- ✓ La creación y edición de los contenidos se realiza a través de un editor web, lo que facilita a las personas sin especialización técnica la actualización del contenido web.
- ✓ La mayoría de los CMS permiten crear grupos de usuarios con diferentes niveles de permisos.
- ✓ Esta flexibilidad permite a los usuarios crear contenidos y nuevas páginas dentro del sitio web sin necesidad de recurrir a un diseñador.
- ✓ La mayoría de los CMS tienen extensiones que añaden potentes características a los sitios web.

2.5 Clasificación de los CMS

A pesar de que los CMS poseen un perfil común y podamos etiquetarlos, algunos están orientados específicamente a un determinado fin y otros sencillamente contemplan otros aspectos, si tenemos en cuenta usar un CMS debemos investigar para asegurarnos de que elegimos aquel que vaya a cubrir nuestras necesidades [18].

- ✓ **Sistemas prioritarios:** Son herramientas creadas a medida para actualizar una página web
- ✓ **Gestión de portales:** Estos CMS sirven para gestionar todo el contenido de un Sitio Web ofreciendo funcionalidad diversa. Uno de los principales es Joomla (<http://www.joomla.org>) que ha ganado amplia popularidad convirtiéndose en el más usado en esta categoría, por sus facilidades para gestionar un Sitio.
- ✓ **Blogs:** Estos permiten la publicación de Noticias en orden cronológico, permitiendo comentarios. Dos de los más populares son WordPress y Blogger.
- ✓ **Gestores de foros:** Estos permiten la discusión en línea de usuarios entre ellos: phpBB, SMF, MyBB.
- ✓ **Wikis:** La Wiki es un tipo especial de Web que permite la creación colaborativa de contenidos, el ejemplo más destacado es la misma Wikipedia, otros gestores de Wiki son: MediaWiki, TikiWiki.
- ✓ **Gestores de comercio electrónico:** Estos permiten generar Sitios Web específicos para comercio electrónico. Ejemplo: osCommerce (licenciaGPL), Dynamicweb eCommerce (comercial).

- ✓ **Galerías:** Estos permiten administrar y generar automáticamente un portal o sitio Web que muestra contenido audiovisual, normalmente imágenes. Ejemplo: Gallery (<http://gallery.menalto.com>) y Dragonfly CMS.
- ✓ **Gestores de E-learning:** Estos son los denominados LMS (Learning management Systems). Las particularidades del proceso de enseñanza aprendizaje han hecho que se desarrollen diferentes tipos de plataformas o LMS, este Diplomado se gestiona desde Moodle (<http://moodle.org/>) que se ha convertido en uno de los más utilizados a nivel mundial. Existen también plataformas especialmente diseñadas teniendo en cuenta las necesidades de las publicaciones digitales, tales como periódicos, revistas, etc. Ejemplo: ePrints, Thinkindot CMS.

Capítulo 3: Tecnología de desarrollo

El desarrollo de aplicaciones web ha evolucionado enormemente en la última década. Se han creado multitud de tecnologías, frameworks de desarrollo de aplicaciones, bibliotecas, aplicaciones configurables, arquitecturas, modelos de publicación de versiones, entre otras. Es conveniente conocer los elementos más importantes utilizados en esta monografía.

3.1 Python

Python es uno de los lenguajes de programación más antiguos, dinámicos y populares que existen, pero no siempre así, el lenguaje se remonta a finales de los 80 y principios de los 90 y su implementación empezó en el año 1989, fue creado por Guido Van Rossum como una actualización del lenguaje ABC, estos dos tienen bastante parecido en la sintaxis [6].

En un principio python no tuvo la popularidad de hoy en día, fue adquiriendo a medida que la comunidad de desarrolladores creaba librerías para ampliar su campo de utilización particularmente de cómputo científico, siendo este su fuerte y la razón por la que hoy en día es tan utilizado.

Python actualmente es utilizado en casi todos los ámbitos, desde programación de microcontroladores hasta desarrollo de inteligencia artificial, en el caso de este proyecto lo utilizamos como lenguaje servidor para el Sistema Manejador de Contenido.

3.1.1 Características del lenguaje

3.1.1.1 Propósito general

Se pueden crear todo tipo de programas. No es un lenguaje creado específicamente para la web, aunque entre sus posibilidades sí se encuentra el desarrollo de páginas.

3.1.1.2 Multiplataforma

Hay versiones disponibles de Python en muchos sistemas informáticos distintos. Originalmente se desarrolló para Unix, aunque cualquier sistema es compatible con el lenguaje siempre y cuando exista un intérprete programado para él.

3.1.1.3 Interpretado

Quiere decir que no se debe compilar el código antes de su ejecución. En realidad, sí que se realiza una compilación, pero esta se realiza de manera transparente para el programador. En ciertos casos, cuando se ejecuta por primera vez un código, se producen unos bytecodes que se guardan en el sistema y que sirven para acelerar la compilación implícita que realiza el intérprete cada vez que se ejecuta el mismo código.

3.1.1.4 Interactivo

Python dispone de un intérprete por línea de comandos en el que se pueden introducir sentencias. Cada sentencia se ejecuta y produce un resultado visible, que puede ayudarnos a entender mejor el lenguaje y probar los resultados de la ejecución de porciones de código rápidamente.

3.1.1.5 Orientado a Objetos

La programación orientada a objetos está soportada en Python y ofrece en muchos casos una manera sencilla de crear programas con componentes reutilizables.

3.1.1.6 Funciones y librerías

Dispone de muchas funciones incorporadas en el propio lenguaje, para el tratamiento de strings, números, archivos, etc. Además, existen muchas librerías que podemos importar en los programas para tratar temas específicos como la programación de ventanas o sistemas en red o cosas tan interesantes como crear archivos comprimidos en .zip.

3.1.1.7 Sintaxis clara

Por último, destacar que Python tiene una sintaxis muy visual, gracias a una notación indentada (con márgenes) de obligado cumplimiento. En muchos lenguajes, para separar porciones de código, se utilizan elementos como las llaves o las palabras clave begin y end. Para separar las porciones de código en Python se debe tabular hacia dentro, colocando un margen al código que iría dentro de una función o un bucle. Esto ayuda a que todos los programadores adopten unas mismas notaciones y que los programas de cualquier persona tengan un aspecto muy similar.

3.2 JavaScript

JavaScript nació bajo la necesidad de tener un lenguaje de programación que manejara los formularios de las páginas web para que así estas no tardaran tanto

en responder, con esto iba a poder ser posible validar información antes de llegar al servidor [8].

Brendan Eich, un programador que trabajaba en Netscape, pensó que podría solucionar este problema adaptando otras tecnologías existentes (como ScriptEase) al navegador Netscape Navigator 2.0, que iba a lanzarse en 1995. Inicialmente, Eich denominó a su lenguaje LiveScript.

Posteriormente, Netscape firmó una alianza con Sun Microsystems para el desarrollo del nuevo lenguaje de programación. Además, justo antes del lanzamiento Netscape decidió cambiar el nombre por el de JavaScript. La razón del cambio de nombre fue exclusivamente por marketing, ya que Java era la palabra de moda en el mundo informático y de Internet de la época.

Actualmente JavaScript es de los lenguajes más usados por la facilidad que nos brinda. Su principal virtud es que se puede ejecutar en el navegador y nos permite el manejo sencillo de efectos, animación, validación, etc., de páginas web y por ende permitiéndonos crear sitios cada vez más dinámicos y cómodos para los usuarios.

3.2.1 Características

3.2.1.1 Imperativo y estructurado

JavaScript es compatible con gran parte de la estructura de programación de C (por ejemplo, sentencias if, bucles for, sentencias switch, etc.). Con una salvedad, en parte: en C, el ámbito de las variables alcanza al bloque en el cual fueron definidas; sin embargo JavaScript no es compatible con esto, puesto que el ámbito de las variables es el de la función en la cual fueron declaradas. Esto cambia con la versión de ECMAScript 2015, ya que añade compatibilidad con block scoping por medio de la palabra clave let. Como en C, JavaScript hace distinción entre expresiones y sentencias. Una diferencia sintáctica con respecto a C es la inserción automática de punto y coma, es decir, en JavaScript los puntos y coma que finalizan una sentencia pueden ser omitidos.

3.2.1.2 Dinámico

Como en la mayoría de lenguajes de scripting, el tipo está asociado al valor, no a la variable. Por ejemplo, una variable x en un momento dado puede estar ligada a un número y más adelante, religada a una cadena. JavaScript es compatible con varias formas de comprobar el tipo de un objeto, incluyendo duck typing. Una forma de saberlo es por medio de la palabra clave typeof.

3.2.1.3 Orientado a Objetos

JavaScript está formado casi en su totalidad por objetos. Los objetos en JavaScript son arrays asociativos, mejorados con la inclusión de prototipos. Las propiedades y sus valores pueden ser creados, cambiados o eliminados en tiempo de ejecución. La mayoría de propiedades de un objeto (y aquellas que son incluidas por la cadena de la herencia prototípica) pueden ser enumeradas a por medio de la instrucción de bucle for... in. JavaScript tiene un pequeño número de objetos predefinidos como son Function y Date.

3.2.1.4 Evaluación en tiempo de ejecución

JavaScript incluye la función eval que permite evaluar expresiones expresadas como cadenas en tiempo de ejecución. Por ello se recomienda que eval sea utilizado con precaución y que se opte por utilizar la función JSON.parse() en la medida de lo posible, pues resulta mucho más segura.

3.2.1.5 Funcional

A las funciones se les suele llamar ciudadanos de primera clase; son objetos en sí mismos. Como tal, poseen propiedades y métodos, como .call() y .bind(). Una función anidada es una función definida dentro de otra. Esta es creada cada vez que la función externa es invocada. Además, cada función creada forma una clausura; es el resultado de evaluar un ámbito conteniendo en una o más variables dependientes de otro ámbito externo, incluyendo constantes, variables locales y argumentos de la función externa llamante. El resultado de la evaluación de dicha clausura forma parte del estado interno de cada objeto función, incluso después de que la función exterior concluya su evaluación.

3.3 FrameWorks

Un framework, entorno de trabajo o marco de trabajo es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar [9].

En el desarrollo de software, un entorno de trabajo es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio, y provee una estructura y una especial metodología de trabajo, la cual extiende o utiliza las aplicaciones del dominio.

3.4 Django

Django es un framework escrito en Python para realizar sitios web, utilizando el Modelo-Vista-Template. A principios fue creado para gestionar varias páginas de noticias de la empresa World Company de Lawrence, fue liberada en julio de 2005.

Django tiene como meta principal facilitar el desarrollo de sitios web complejos haciendo énfasis en la reutilización de código, la conectividad y extensiones de componentes.

Los orígenes de Django en la administración de páginas de noticias son evidentes en su diseño, ya que proporciona una serie de características que facilitan el desarrollo rápido de páginas orientadas a contenidos. Por ejemplo, en lugar de requerir que los desarrolladores escriban controladores y vistas para las áreas de administración de la página, Django proporciona una aplicación incorporada para administrar los contenidos, que puede incluirse como parte de cualquier página hecha con Django y que puede administrar varias páginas hechas con Django a partir de una misma instalación; la aplicación administrativa permite la creación, actualización y eliminación de objetos de contenido, llevando un registro de todas las acciones realizadas sobre cada uno, y proporciona una interfaz para administrar los usuarios y los grupos de usuarios (incluyendo una asignación detallada de permisos) [3].

3.5 jQuery

jQuery es una biblioteca JavaScript rápida, pequeña y rica en funciones. Hace cosas como el desplazamiento de documentos HTML y manipulación, manejo de eventos, animación y peticiones Ajax mucho más simple con una API fácil de usar que funciona a través de una multitud de navegadores. Con una combinación de versatilidad y extensibilidad [12].

Comenzar con jQuery puede ser fácil o desafiante, dependiendo de su experiencia con JavaScript, HTML, CSS y conceptos de programación en general.

Una cosa importante a saber es que jQuery es sólo una biblioteca de JavaScript. Todo el poder de jQuery se accede a través de JavaScript, por lo que tener una fuerte comprensión de JavaScript es esencial para la comprensión, estructuración

y depuración de su código. Mientras que el trabajo con jQuery regularmente puede, con el tiempo, mejorar su habilidad con JavaScript, puede ser difícil empezar a escribir jQuery sin un conocimiento práctico de las construcciones integradas de JavaScript y la sintaxis.

3.6 Bootstrap

Un entorno de desarrollo con una serie de recursos que simplifican el desarrollo de un proyecto web con html5, css3 y jQuery, de manera que simplifica mucho el trabajo a la hora de diseñar, ya que el framework bootstrap ya tiene una buena parte del trabajo hecho lo cual simplifica mucho la tarea del desarrollo [13].

Fue creado por Twitter en un intento por unificar todo su trabajo, pero se dieron cuenta que podría ser utilizado por otras personas, así que decidieron publicarlo como una librería que nos ayudaría a desarrollar un sitio web responsivo con mucha mayor facilidad.

3.6.1 Ventajas de usar Bootstrap

- ✓ Puedes tener una web bien organizada de forma visual rápidamente: la curva de aprendizaje hace que su manejo sea asequible y rápido si ya sabes maquetar.
- ✓ Permite utilizar muchos elementos web: desde iconos a desplegables, combinando HTML5, CSS y Javascript.
- ✓ Sea lo que sea que creemos, el diseño será adaptable, no importa el dispositivo, la escala o resolución.
- ✓ El grid system: maquetar por columnas nunca fue tan fácil. Además, son muy configurables.
- ✓ Se integra muy bien con las principales librerías Javascript.
- ✓ El haber sido creado por Twitter nos da ciertas garantías: está muy pensado y hay mucho trabajo ya hecho. Por lo tanto, hay una comunidad muy activa creando, arreglando cosas, ofreciendo plugins y mucho más.
- ✓ Cuenta con implementaciones externas para WordPress, Drupal, etc. Nos permite usar preprocesadores de css, para enriquecer aún más los estilos de la web.

Capítulo 4: Metodologías de desarrollo ágil

Gallo & Vergara, (2009), mencionan que las metodologías ágiles o “ligeras” constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de e-projects que las metodologías convencionales (ISO-9000, CMM, etc.) debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración.

Existen muchos métodos de desarrollo ágil, donde la mayoría trata de minimizar los riesgos desarrollando software en cortos lapsos de tiempo. El software desarrollado en una unidad de tiempo es llamado un sprint, la cual debe durar de una a cuatro semanas. Cada sprint del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación. Un sprint no debe agregar demasiada funcionalidad para justificar el lanzamiento del producto al mercado, pero la meta es tener un demo (sin errores) al final de cada sprint. Al final de cada sprint el equipo vuelve a evaluar las prioridades del proyecto.

Los métodos ágiles también enfatizan que el software funcional es la primera medida del progreso, combinado con la preferencia por las comunicaciones cara a cara; generalmente los métodos ágiles son criticados y tratados como "indisciplinados" por la falta de documentación técnica. (Benedicto, 2009).

Muchos métodos similares al ágil fueron creados antes del 2000. Entre los más notables se encuentran: Scrum (1986), Crystal Clear (cristal transparente), programación extrema o XP (1996), desarrollo de software adaptativo, feature driven development, Método de desarrollo de sistemas dinámicos (1995). Kent Beck creó el método de Programación Extrema (usualmente conocida como XP) en 1996 como una forma de rescatar el proyecto del Sistema exhaustivo de compensaciones de Chrysler (C3). Mientras Chrysler cancelaba ese proyecto, el método fue refinado por Ron Jeffries.

4.1 Metodología SCRUM

La Metodología Scrum es un proceso de desarrollo de software iterativo y creciente utilizado, comúnmente, en entornos basados en el desarrollo ágil de software. Scrum es un framework de desarrollo ágil de software. El trabajo es estructurado en ciclos de trabajo llamados Sprints, iteraciones de trabajo con una duración típica de dos a cuatro semanas. Durante cada sprint, los equipos eligen de una lista de requerimientos de cliente priorizados, llamados historias de usuarios, para que las características que sean desarrolladas primero sean las de

mayor valor para el cliente. Al final de cada sprint, se entrega un producto potencialmente lanzable/distribuible/comerciable.

Scrum se caracteriza por ser un modelo que define un conjunto de prácticas y roles que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto. Los roles principales en Scrum son el Scrum Master, el Product Owner, y el Equipo Scrum.

Las características más marcadas que se logran notar en Scrum serían:

- ✓ Gestión regular de las expectativas del cliente.
- ✓ Resultados anticipados.
- ✓ Flexibilidad y adaptación.
- ✓ Retorno de inversión.
- ✓ Mitigación de riesgos.
- ✓ Productividad y calidad.
- ✓ Alineamiento entre cliente y equipo.
- ✓ Un equipo motivado.

4.1.1 Actividades a realizar

4.1.1.1 Sprint Planning

La planificación de las tareas a realizar en la iteración se divide en dos partes:

Primera parte de la reunión: Se realiza en un Timebox de cómo máximo 4 horas:

- ✓ El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto, pone nombre a la meta de la iteración (de manera que ayude a tomar decisiones durante su ejecución) y propone los requisitos más prioritarios a desarrollar en ella.
- ✓ El equipo examina la lista, pregunta al cliente las dudas que le surgen, añade más condiciones de satisfacción y selecciona los objetivos/requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.

Segunda parte de la reunión: Se realiza en un timebox de cómo máximo 4 horas. El equipo planifica la iteración, elabora la táctica que le permitirá conseguir el mejor resultado posible con el mínimo esfuerzo. Esta actividad la realiza el equipo dado que ha adquirido un compromiso, es el responsable de organizar su trabajo y es quien mejor conoce cómo realizarlo.

- ✓ Define las tareas necesarias para poder completar cada objetivo/requisito, creando la lista de tareas de la iteración (Sprint backlog) basándose en la definición de completado.

- ✓ Realiza una estimación conjunta del esfuerzo necesario para realizar cada tarea.
- ✓ Cada miembro del equipo se auto asigna a las tareas que puede realizar.

4.1.1.2 Sprint

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas). Cada sprint tiene que proporcionar un resultado completo, un incremento de producto que sea susceptible de ser entregado con el mínimo esfuerzo cuando el cliente lo solicite.

4.1.1.3 Scrum Daily Meeting

El objetivo de esta reunión es facilitar la transferencia de información y la colaboración entre los miembros del equipo para aumentar su productividad, al poner de manifiesto puntos en que se pueden ayudar unos a otros. Cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo del sprint, obstáculos que pueden impedir este objetivo) para al finalizar la reunión poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso conjunto que el equipo adquirió para el sprint.

4.1.1.4 Sprint Review

Reunión informal donde el equipo presenta al cliente los requisitos completados en el sprint, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo, haciendo un recorrido por ellos lo más real y cercano posible al objetivo que se pretende cubrir.

4.1.1.5 Sprint Retrospective

Con el objetivo de mejorar de manera continua su productividad y la calidad del producto que está desarrollando, el equipo analiza cómo ha sido su manera de trabajar durante el sprint, por qué está consiguiendo o no los objetivos a que se comprometió al inicio del sprint y por qué el incremento de producto que acaba de demostrar al cliente era lo que él esperaba o no.

4.1.1.6 Los roles del equipo

Los roles asignados son los siguientes:

- ✓ **Scrum Master:** Es la persona que se encargará de coordinar el equipo y asignar las tareas a realizar.
- ✓ **Product Owner:** Son los grupos de interés a los que va dedicado el proyecto/producto/servicio que se está desarrollando. Son los que dicen

qué es lo que se quiere hacer y cuáles son los objetivos. En el caso de no estar presentes, se debe nombrar un representante de fuera del equipo que se encargue de defender sus intereses y su punto de vista.

- ✓ **Scrum Team:** Son los responsables de desarrollar las tareas. Se recomienda crear equipos no muy grandes (menos de 10 personas) donde las personas se complementen, de forma que cada uno tenga unos conocimientos específicos y unas actividades pre asignadas acordes con estos.
- ✓ **Los consumidores o usuarios (Customers):** Son los que usarán el producto final. Muchas veces se confunden con los clientes, pero no son los mismos. Hablando claro: “cliente es el que paga (y por lo tanto decide) y consumidor el que usa el producto”. A veces cliente y consumidor son la misma persona, pero otras veces no.

4.1.1.7 Product BackLog

La lista de objetivos/requisitos priorizada representa la visión y expectativas del cliente respecto a los objetivos y entregas del producto o proyecto. El cliente es el responsable de crear y gestionar la lista (con la ayuda del Facilitador y del equipo, quien proporciona el coste estimado de completar cada requisito). Dado que reflejar las expectativas del cliente, esta lista permite involucrarle en la dirección de los resultados del producto o proyecto. Contiene los objetivos/requisitos de alto nivel del producto o proyecto, que se suelen expresar en forma de historias de usuario. Para cada objetivo/requisito se indica el valor que aporta al cliente y el coste estimado de completarlo. La lista está priorizada balanceando el valor que cada requisito aporta al negocio frente al coste estimado que tiene su desarrollo, es decir, basándose en el Retorno de la Inversión (ROI).

En la lista se indican las posibles iteraciones y las entregas (releases) esperadas por el cliente (los puntos en los cuales desea que se le entreguen los objetivos/requisitos completados hasta ese momento), en función de la velocidad de desarrollo de los equipos que trabajarán en el proyecto. Es conveniente que el contenido de cada iteración tenga una coherencia, de manera que se reduzca el esfuerzo de completar todos sus objetivos.

4.1.1.8 ScrumTaskboard

La lista de objetivos a completar en el sprint se puede gestionar mediante un tablón de tareas (ScrumTaskboard). Al lado de cada objetivo se ponen las tareas necesarias para completarlo, en forma de post-its, y se van moviendo hacia la derecha para cambiarlas de estado (pendientes de iniciar, en progreso, hechas). Para cada miembro del equipo se puede utilizar adhesivos de colores más pequeños sobre cada tarea, de manera que se pueda ver en qué tareas está trabajando cada cual.

4.1.1.9 Burndown Chart

Es un gráfico de trabajo pendiente a lo largo del tiempo muestra la velocidad a la que se está completando los objetivos/requisitos. Permite extrapolar si el equipo podrá completar el trabajo en el tiempo estimado.

4.2 Metodología Extreme Programming (XP) o Programación Extrema

Kendall & Kendall (2005), encontró que la Metodología ágil basada en cuatro principios: simplicidad, comunicación, retroalimentación y valor. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

Las características de la metodología XP son de desarrollo iterativo e incremental y pruebas continuas. Se aconseja escribir el código de la prueba antes de la codificación.

- ✓ **Programación en parejas:** Se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la calidad del código escrito de esta manera, es revisada y discutida mientras se escribe y es más importante que la posibilidad de pérdida de la productividad inmediata.
- ✓ **Propiedad del código compartida:** En vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.
- ✓ **Simplicidad en el código:** Es la mejor manera de que las cosas funcionen, cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

4.2.1 Ventajas y desventajas de la metodología XP

4.2.1.1 Ventajas

- ✓ Programación organizada.
- ✓ Menor tasa de errores.
- ✓ Satisfacción del programador.

4.2.1.2 Desventajas

- ✓ Es recomendable emplearlo solo en proyectos a corto plazo.
- ✓ Altas comisiones en caso de fallar.

4.2.1.3 Beneficios

- ✓ El cliente tiene el control sobre las prioridades.
- ✓ Se hacen pruebas continuas durante el proyecto.
- ✓ La XP es mejor utilizada en la implementación de nuevas tecnologías donde los requerimientos cambian rápidamente.

4.2.2 Ciclo de vida de la metodología XP

El ciclo de vida de XP se enfatiza en el carácter interactivo e incremental del desarrollo, es una iteración de desarrollo es un período de tiempo en el que se realiza un conjunto de funcionalidades determinadas que en el caso de XP corresponden a un conjunto de historias de usuarios.

Kendall (2005), se menciona que las iteraciones son relativamente cortas porque se piensa que entre más rápido se le entregue el desarrollo al cliente más retroalimentación se va a obtener, esto va a representar una mejor calidad del producto a largo plazo.

- ✓ **Fase de la exploración:** En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología.
- ✓ **Fase de la planificación:** Se priorizan las historias de usuario y se acuerda el alcance del release. Los programadores estiman cuánto esfuerzo requiere cada historia y a partir de allí se define el cronograma. La duración del cronograma del primer reléase no excede normalmente dos meses. La fase de planeamiento toma un par de días. Se deben incluir varias

iteraciones para lograr un release. El cronograma fijado en la etapa de planeamiento se realiza a un número de iteraciones, cada una toma de una a cuatro semanas en ejecución.

Capítulo 5: Elección de Metodología

Se determina usar una metodología ágil de acuerdo a los resultados que se espera obtener y el alcance que se quiere llegar con el proyecto. En la cual dicha elección estuvo enmarcada en las necesidades de la empresa las cuales fueron:

- ✓ Gestión regular de las expectativas del cliente.
- ✓ Resultados anticipados.
- ✓ Flexibilidad y adaptación.
- ✓ Retorno de inversión.
- ✓ Mitigación de riesgos.
- ✓ Productividad y calidad.
- ✓ Alineamiento entre cliente y equipo.

Se compararon las metodologías ágiles XP con Scrum ya que dentro de las metodologías ágiles son las más desatadas dentro del mercado y adicionalmente son las metodologías con las que el equipo de desarrollo se encuentra más familiarizado.

Se elaboró un cuadro de comparación de metodologías el cual se realizó bajo los siguientes postulados:

- ✓ E1: Enfoque de la Metodología.
- ✓ E2: Equipos de trabajo.
- ✓ E3: Duración de las tareas o actividades.
- ✓ E4: Resultado de las tareas o actividades.
- ✓ E5: Forma de trabajo de la metodología.

Comparación de metodologías ágiles SCRUM y XP

POSTULADO	SCRUM	XP
E1	Es una metodología de desarrollo ágil basada en la administración del proyecto.	Es una metodología de desarrollo que está más centrada en la programación o creación del producto.

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

E2	Cada miembro de del equipo trabaja de forma individual.	Los miembros del equipo programan en parejas.
E3	Las iteraciones de entrega son de 1 a 4 semanas.	Las iteraciones de entrega son de 1 a 3 semanas.
E4	Al finalizar un Sprint, las tareas del Sprint Backlog que se hayan realizado y que el Product Owner (propietario del producto) haya mostrado su conformidad ya no se retoca. Si funciona y está bien, se aparta y a otra cosa.	Las tareas se van terminando, aunque son susceptibles de ser modificadas durante el transcurso de proyecto, incluso, después de que funcionen correctamente.
E5	Trata de seguir el orden de prioridades que marca el Product Owner en el Sprint Backlog pero puede cambiarlo si es mejor para el desarrollo de la tareas.	El equipo de desarrollo sigue estrictamente el orden de prioridad de las tareas definido por el cliente.

Tabla N° 1. Comparación de metodologías ágiles SCRUM y XP

Los criterios que se muestra en la Tabla han sido colocados dependiendo a lo que se quiere lograr con este proyecto, por lo criterios de la elección de la metodología se enfatizan en estos puntos, para así poder ver si es factible o si cumplió con lo que se establece. En la siguiente se indica el nivel de soporte para la utilización de la metodología la cual se mostrará la escala de los puntajes.

Escala de puntajes de Metodología

PUNTAJE	CONCEPTO
1	Muy poco
2	Poco
3	Normal
4	Bueno
5	Muy Bueno

Tabla N° 2. Escala de puntajes de Metodología

En la Tabla siguiente se muestra un comparativo de las metodologías presentadas según criterios que se evaluaron de acuerdo al proyecto.

Tabla de Comparación de Metodologías

POSTULADO	SCRUM	XP
E1	5	2
E2	3	4
E3	4	5
E4	4	5
E5	5	4
TOTAL	21	20

Tabla N° 3. Tabla de Comparación de Metodologías

La metodología que se utiliza en este proyecto, es la metodología SCRUM con un puntaje de 21 la cual es mayor a la otra metodología que se encuentran en la tabla lo cual indica es la que apoyaría en grandes proporciones al desarrollo de este proyecto.

De todas las metodologías, ésta es la que ha recibido más atención. Esto se debe a la habilidad de atraer a las personas a este acercamiento, y tomar un papel principal en él. De algunas maneras la popularidad de SCRUM ha acaparado la atención de las otras metodologías y sus valiosas ideas.

Capítulo 6: Elección de Tecnologías de Desarrollo

Uno de los pasos más importantes para la creación del Sistema Manejador de Contenido Web fue la elección de los lenguajes, FrameWorks y tecnologías que se usaran para el desarrollarlo. Entre los puntos notables para la elección de las tecnologías se encuentran:

- ✓ Software Libre.
- ✓ Multiplataforma.

Listaremos cada herramienta utilizada y el objetivo por el cual fue incluida en el desarrollo.

FrontEnd: Sin mucho que comparar se seleccionó a JavaScript como lenguaje

para el FrontEnd de la aplicación, al ser el lenguaje por excelencia de los lenguajes del lado del cliente. En esta sección nos auxiliamos de jQuery una librería de JavaScript que nos permite entre otras cosas compatibilidad con navegadores y un conjunto de funciones que añaden simplicidad al código escrito.

BackEnd: Para el desarrollo del BackEnd se utilizó Django, Framework de Python, que entre características notables se encuentra la multiplataforma y licencia de software libre.

Django al usar la metodología de desarrollo Modelo Vista Template (MVT) y se selecciona por encima de otras herramientas ya que no poseen un set tan amplio de componentes que nos brinden:

- ✓ ORM fácil, super intuitivo, y poderoso.
- ✓ Librerías de terceros bastante poderosas, como Django REST Framework ("djanoorestframework"), ideal para desarrollar puntos REST.
- ✓ Sistema muy cómodo de MVT y direccionamiento y referenciamiento de URLs de manera "cómoda".

Al ser basado en Python y orientado a proceso en lugar de a petición, se puede implementar un caché para los entornos de desarrollo sin obligarles a instalar un Memcached, hasta la etapa de producción o staging. De la misma forma, se permite hacer una única carga pesada de todo el esquema de clases (cosa que sería peligrosamente pesada si se hiciera petición por petición), lo que permite hacer un ORM sumamente poderoso.

Para la base de datos se eligió el lenguaje SQL ya que este sistema era ideal para una base de datos relacional y se escogió MYSQL como motor de base de datos ya que al ser gratuito y multiplataforma este ya cumplía con dos de los objetivos más importantes.

A Continuación, se mostrará una tabla con los requisitos que deben cumplir cada una de las tecnologías elegidas y las descartadas

<i>Tecnología/ Lenguaje</i>	<i>Amigable</i>	<i>Soporte</i>	<i>Fácil administración</i>	<i>Open source</i>	<i>Multiplatafo rma</i>
<i>JavaScript</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>jQuery</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>Bootstrap</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>Python</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>Django</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>PHP</i>	<i>No</i>	<i>Si</i>	<i>no</i>	<i>si</i>	<i>si</i>

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

<i>Flask</i>	<i>No</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>ASP.NET</i>	<i>No</i>	<i>Si</i>	<i>no</i>	<i>no</i>	<i>no</i>
<i>MYSQL</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>si</i>	<i>si</i>
<i>SQL Server</i>	<i>Si</i>	<i>Si</i>	<i>si</i>	<i>no</i>	<i>no</i>

Tabla N° 4. Comparativa de requisitos solicitados en las tecnologías de desarrollo

Capítulo 7: Información SAREC

7.1 Historia

El Programa de Investigación en Tecnologías de Información y Comunicaciones (PROITIC), inició en 1992 en la Facultad de Electrotecnia y Computación (FEC) de la Universidad Nacional de Ingeniería (UNI), Nicaragua, auspiciado por la Agencia Sueca para el Desarrollo Internacional (ASDI). Los contrapartes en Suecia de este Programa fueron el Departamento de Sistemas de Información y Control Industrial, Departamento de Radiocomunicación ambos del Real Instituto Tecnológico de Estocolmo (KTH), Suecia, y el Departamento de Electrociencia en el Instituto Tecnológico de Lund (LTH) , Suecia, que se integró a partir de 1999.

El objetivo general del Programa es fortalecer la capacidad de investigación e innovación en la Universidad Nacional de Ingeniería en el área de Tecnologías de la Información y Comunicaciones (TIC) con un cercano vínculo con los actores nacionales, centroamericanos e internacionales, fortaleciendo de esta manera la relevancia y calidad de los resultados del proceso de investigación científica y su aplicación.

El Programa condujo a los primeros profesores involucrados de parte de la FEC, UNI, a la obtención de grados académicos de Doctorado (PhD) o el Teknologie Licentiate. Este último es un grado académico en los países nórdicos entre el MSc. y el Doctorado (PhD) conocido en algunos círculos como MPhil. Este Programa se desarrolló a través de visitas a Suecia y a los países en donde se publican los resultados de investigación (conferencias, simposios científicos, etc.) con visitas de los colegas de investigación de Suecia a Nicaragua. Esto ha permitido que los investigadores nicaragüenses y suecos tengan una más cercana visión de la aplicación de los temas de investigación a la realidad y contexto nacional.

Los miembros de este Programa han desarrollado vínculos activos con redes de investigación, sociedades académicas y profesionales pudiendo compartir el conocimiento en las diferentes áreas TIC que se desarrollan, publicando internacionalmente y con experiencia en servicios, consultorías y casos de estudios con empresas nacionales e internacionales en Nicaragua. Se cuenta con una infraestructura básica para el desarrollo de actividades de investigación, producción científica mediante publicación de resultados de investigación en revistas de prestigio internacional con arbitraje y la participación en congresos, seminarios y simposios nacionales e internacionales con arbitraje, así como la participación de investigadores del programa como miembros de comités de revisores para congresos nacionales e internacionales.

7.2 Organización

El programa de investigación en TIC de la Universidad Nacional de Ingeniería (Nicaragua), consta de una estructura organizativa jerárquica como se presenta en el siguiente diagrama.

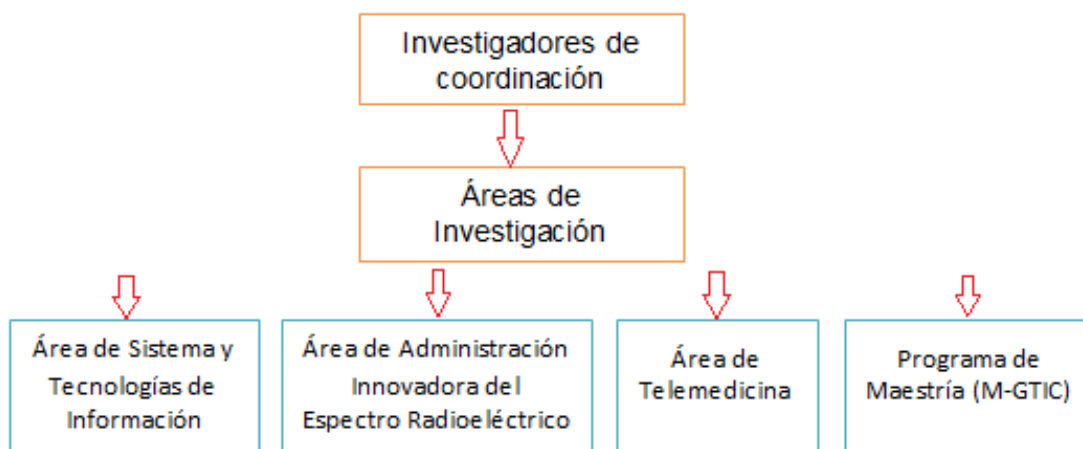


Figura N° 7. Organización del Departamento de Investigación en TIC

Descripción breve de cada estructura organizativa del Programa de Investigación en TIC de la Universidad Nacional de Ingeniería (Nicaragua):

✓ **Investigadores de coordinación**

Es la primera línea de autoridad del programa. Son los encargados de definir estrategias y aprobar presupuestos.

✓ **Áreas de Investigación**

Son las que llevan a cabo o que participan en una investigación, es decir, que lleva adelante un proyecto orientado a la búsqueda de conocimiento y al esclarecimiento de hechos y de relaciones.

En las siguientes tablas se muestra la información de la distribución del personal administrativo, fue recopilada directamente dentro de la estructura organizativa del Programa de Investigación en TIC de la Universidad Nacional de Ingeniería (Nicaragua).

Tabla N° 5: Distribución de personal en las estructuras administrativas.

Descripción	Cantidad
Investigadores de coordinación	1
Áreas de Investigación no tenemos 17 areas???	17
Total	18

Tabla N° 6: Distribución Docente por Categoría

Descripción	Cantidad
Investigadores con postgrado	12
Investigadores PhD	4
Investigadores PhD students en Mondragon University, Spain	2
Total	18 (están seguros??)

7.3 Líneas de Investigación

A continuación, se presenta información sobre las líneas de investigación del Programa de Investigación en TIC de la Universidad Nacional de Ingeniería (Nicaragua):

Descripción	Cantidad
Sistemas y Tecnologías de Información	12
Sistema de Radiocomunicaciones	4
Telemedicina (aplicaciones médicas de procesamiento de señales)	2
Programa de Maestría en Gestión TIC	
Total	18

7.3.1 Sistemas y tecnologías de información

Esta área tiene como foco de investigación la Arquitectura de Sistemas de Información o Tecnologías de Información en Organizaciones. Este tema de estudio analiza y propone modelos y métodos para la identificación, definición y evolución de los sistemas de información de una manera integral con alineación con las metas de la organización y sus procesos. Cada vez es más obvio que los sistemas de información (generalmente basados en computadoras, redes y acceso a Internet entre otras características o entidades) deben estar más estrechamente ligados con las estrategias, misión y visión de la organización que apoya, en especial los procesos a los que los sistemas de información que le apoyan. Cualquier cambio de una parte de la organización o del sistema de información influye en los otros módulos o recursos del resto del sistema, de manera que a mediano y largo plazo es imposible manejar la evolución del sistema de información y la organización que apoya de manera individual o separadamente.

7.3.2 Sistemas de Radiocomunicaciones.

Esta área tiene como antecedentes el hecho de que Nicaragua es un país que ha enfrentado varios eventos negativos que han tenido un impacto devastador en la infraestructura de telecomunicaciones disponible. A saber, en 1972 un terremoto destruyó totalmente la ciudad capital Managua. A partir de entonces se inició una severa degradación de la economía que condujo a un serio daño a la infraestructura de comunicaciones. Este proceso de degradación continuó durante la década de los ochenta, principalmente debido a la guerra y la poca inversión en esta área.

El resultado de estos eventos es una pobre o inexistente infraestructura de telecomunicaciones en las áreas urbanas y rurales. Sin embargo, durante los últimos años ha habido un progreso con el despliegue de redes celulares inicialmente dirigidas a servicios de voz. Nuevos servicios incluyendo la transmisión de datos están comenzando a ser ofrecidos a partir de sistemas celulares GSM, GPRS y cdma2000. No obstante, la infraestructura de red celular en Nicaragua está disponible en las áreas urbanas (principales ciudades y carreteras) mientras en las áreas rurales no se tienen acceso a ningún tipo de servicios. En tales áreas el problema no es tecnológico (e.g. existe la posibilidad de utilizar comunicaciones por satélite) sino lo remoto de su ubicación y la pobreza

de su población. Esto nos lleva a la siguiente pesquisa, ¿Es posible proveer redes de datos inalámbricas de bajo costo y fácil despliegue para comunicaciones de áreas rurales?

7.3.3 Telemedicina (aplicaciones médicas de procesamiento de señales)

Esta área tiene como foco de investigación el procesamiento de señales como una tecnología, que ha encontrado su aplicación en disciplinas tan diversas como comunicaciones, medicina, ecología y la tecnología del entretenimiento. Algoritmos avanzados del procesamiento de señales constituyen hoy en día una parte integral de muchos productos de consumo, por ejemplo: sistemas de audio, televisión, prótesis de oído, automóviles y teléfonos móviles.

El proyecto de investigación está dirigido hacia la aplicación del procesamiento de señales en las telecomunicaciones. Este estudia y analiza modelos de canal para sistemas de radiocomunicación móvil usando técnicas de procesamiento de señales. La técnica de procesamiento de señales utilizada es la medición en tiempo real de la respuesta al impulso del canal de radiocomunicación para condiciones estacionarias y no estacionarias, en ambientes internos y externos de áreas urbanas.

La principal actividad de investigación está enfocada en la modelación, simulación y medición de canales de comunicación aplicados al diseño de sistemas de radiocomunicación móvil en la banda de frecuencia de los Gigahertz, utilizados en ambientes internos y externos de áreas urbanas, usando técnicas de procesamiento de señales.

7.3.4 Programa de Maestría en Gestión TIC

La Maestría en Gestión TIC (Maestría GTIC) es el enlace natural común entre las actuales áreas de investigación del Programa de investigación TIC consolidando los resultados de los avances en cada área de investigación en marco transdisciplinario. La Maestría GTIC está diseñada para ofertar dos especialidades o menciones que consisten en Tecnologías de Información o Tecnologías de Telecomunicaciones.

Este programa de maestría tiene como objetivo formar profesionales especializados en Tecnologías de Información y Comunicación con conocimientos avanzados en el análisis, evaluación, planificación, ejecución y gestión de proyectos en el ámbito de las TIC requeridos por las instituciones públicas y

privadas de Nicaragua.

7.4 Publicaciones

Descripción	Cantidad
“PhD Dissertations”/ Tesis Doctorales	4
“Licentiate Dissertations” / Tesis Teknologie Licentiate	12
Publicaciones en revistas internacionales (Research papers in peerreviewed international journals)	5
Patentes: Número de patentes licenciadas (1)	1
Publicaciones en conferencias internacionales (Research papers in peerreviewed international proceedings)	45
Publicaciones en revistas nacionales arbitrarias (Research papers in peerreviewed national journals)	5
Publicaciones en revistas nacionales e internacionales de estudiantes del programa de maestría en gestión TIC	5
Supervisión a Tesis de Maestrías	6
Supervisión de Muestras de tesis de Grado	16
Total	99

Tabla Nº 7: Publicaciones Agrupadas por Categorías

7.5 Problemas identificados relacionados con el manejo de la información de PROTIC

A continuación, se presenta algunos problemas identificados sobre el manejo de la información en la página web PROTIC.

- ✓ Dificultad en el mantenimiento de las publicaciones realizadas en el sitio.
- ✓ No existe control del material o capacidad de manipulación de los datos publicados.
- ✓ Contenido web estático y sin una estructura de navegación.
- ✓ No existe una base de datos que mantenga centralizada toda la información sobre los trabajos realizados por el departamento.

7.6 Analizando la información recopilada

Toda institución que posea un sitio de información web debe permitir la creación

y administración del contenido por parte de cualquier integrante de la organización mediante permisos y roles.

El Sitio Web PROITIC es un sistema de información estático el cual la información plasmada no es leída de una base de datos, esto dificulta la capacidad de proporcionar soluciones organizativas y administrativas a los retos del entorno. Cabe señalar que los artículos publicados no poseen una relación entre si ya que no son accesibles o no hacen referencia a los demás artículos mediante menús o sugerencias.

En PROITIC es necesario integrar los componentes del Sitio Web y que cada miembro del programa sea responsable de agregar y actualizar información que le competa. De esa manera se creará un Sitio Web Dinámico que crezca a medida que crece la información del programa.

Capítulo 8: Sistema Manejador de Contenido Web para el Programa de Investigación en TIC

Como fase preliminar a la aplicación de la metodología de desarrollo ágil SCRUM en el diseño del Sistema Manejador de Contenido Web, se desarrolló una serie de reuniones en las cuales se ha definido a los integrantes del proyecto y un alcance preliminar del mismo, con el fin de sentar las bases para la implantación controlada del sistema.

El propósito de esta etapa fue identificar y establecer las necesidades generales del departamento, así como las actividades cotidianas del equipo.

Acá se ha establecido el cronograma inicial, en base al cual se desarrolló el proyecto, definiéndose un total de 18 semanas desde el establecimiento de interesados hasta el desarrollo del presente documento monográfico.

8.1 Reunión de planificación de Sprint

El trabajo que realizar en cada Sprint se previó en la Reunión de Planificación del Sprint. Este plan se crea con la colaboración de todo el Equipo Scrum.

En cada reunión se definió la funcionalidad en el incremento planeado y cómo el equipo de desarrollo creará este incremento. La reunión de planificación de Sprint tradicionalmente respondió la siguiente pregunta:

¿Qué va a ser entregado en el incremento resultante del próximo Sprint?

Aquí se fue definiendo el **Product Backlog** que es el conjunto de requisitos denominados historias, estos son descritos en un lenguaje no técnico. Los requisitos y prioridades se revisaron y ajustaron durante el curso del proyecto a intervalos regulares.

8.2 El Scrum Diario

Acá se define el **Sprint Backlog** que no es más que una lista de las tareas necesarias para llevar a cabo las **historias** del sprint.

Fueron eventos de 15 minutos, cuyo objetivo es que el equipo de desarrollo sincronice actividades, y cree un plan para las próximas 24 horas. Esto se realiza

mediante la inspección del trabajo desde el último Scrum Diario, y la previsión del trabajo que se puede hacer antes del próximo.

El equipo de desarrollo utilizó el Scrum Diario para evaluar el progreso hacia la meta del Sprint y evaluar la tendencia del progreso en finalizar el trabajo en el Sprint Backlog.

Un beneficio adicional es que los Scrums diarios eliminaron otras reuniones, identificaron y eliminaron obstáculos para el desarrollo. Esta es una reunión clave de inspección y adaptación.

Al igual que en el paso anterior se trataba de responder una pregunta:

¿Cómo se va a realizar el trabajo seleccionado?

8.3 Trabajo de desarrollo durante el Sprint

El desarrollo se realizaba de forma iterativa e incremental. Cada iteración, denominada Sprint, tuvo una duración preestablecida de entre 1 y 3 semanas, obteniendo como resultado una versión del software con nuevas funcionalidades listas para ser usadas. En cada nuevo Sprint, se fue ajustando la funcionalidad ya construida y se añaden nuevas funcionalidades priorizándose siempre aquellas que aporten mayor valor al CMS.

Cuando cada sprint estuvo en curso, se aseguró que:

- No se realizan cambios que afectan al objetivo del Sprint.
- No disminuyen los objetivos de calidad.
- El Alcance podrá aclararse y renegociarse entre el departamento (cliente) y el equipo de desarrollo.

8.4 Revisión del Sprint

Se llevo a cabo al final del Sprint, para inspeccionar el incremento y adaptar, si es necesario el **Product Backlog**. La presentación del incremento está destinada a obtener retroalimentación.

La revisión de Sprint incluía los siguientes elementos:

- ✓ El cliente identificaba lo que se ha "hecho" y lo que no se ha "hecho".

- ✓ El equipo de desarrollo discutía lo que anduvo bien durante el Sprint, qué problemas hubo y cómo se resolvieron.
- ✓ El equipo de desarrollo demuestra el trabajo que se ha "hecho" y responde preguntas sobre el Incremento.
- ✓ El cliente analiza el estado actual del **Product Backlog**, y estima fechas de finalización basado en el progreso hasta la fecha.
- ✓ Se hace una revisión de la línea de tiempo, capacidades potenciales para la próxima entrega prevista del producto.

8.5 Retrospectiva del Sprint

Fue una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y crear un plan de mejoras para ejecutar durante el siguiente sprint. El propósito de la retrospectiva de Sprint es:

- ✓ Revisar cómo fue el último Sprint en lo que respecta a las personas, relaciones, procesos y herramientas.
- ✓ Identificar y ordenar los temas principales que salieron bien y las potenciales mejoras.
- ✓ Crear un plan para la implementación de mejoras con respecto a cómo el Equipo Scrum hace su trabajo.

8.6 Roles del proyecto

Al utilizar la metodología Scrum, el equipo se focaliza en construir software de calidad. La gestión de un proyecto Scrum se centra en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

Definición de roles del proyecto	
Scrum Master	Johnny Flores Delgadillo
Product Owner	Alvaro Javier Romero Parrales

Equipo	Alvaro Javier Romero Parrales Sergio Israel Polanco Pao
---------------	--

Tabla Nº 8: Definición de Roles del Proyecto

Scrum Master: Se encargará de administrar el proceso del proyecto, su planificación, coordinación con el equipo y realizar un seguimiento e informes del progreso del proyecto, en términos de calidad y plazos de entrega.

- ✓ Realiza la planificación de las actividades generales del proyecto.
- ✓ Acepta o rechaza los resultados del trabajo del equipo.
- ✓ Remueve impedimentos.
- ✓ Se asegura de que el equipo es completamente funcional y productivo.
- ✓ Permite la estrecha cooperación en todos los roles y funciones.

Product Owner: Se encargará de crear la lista de funcionalidades del sistema, planificar el inicio de cada sprint y la revisión del producto al término de cada sprint para determinar si se cumplió con todas las funcionalidades.

Equipo: Las principales funciones son:

- ✓ Comprometerse al inicio de cada sprint desarrollar todas las funcionalidades en
- ✓ el tiempo determinado.
- ✓ Son responsables de entregar un producto a cada término del Sprint.
- ✓ Definir que se desarrolla del sistema.

8.7 Requerimientos del sistema

Es necesario tener presente que en esta etapa se definieron los requerimientos preliminares que sentaron las bases para la implementación de las etapas de SCRUM en el desarrollo del Sistema Manejador de Contenido Web.

Requerimientos Funcionales	Requerimientos no Funcionales
Creación de una página web para los lectores.	Interfaz intuitiva y amigable

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Desarrollo de un Sistema Manejador de Contenido Web que administre el contenido de la página web.	Alvaro Javier Romero Parrales
Acceso al Sistema Manejador de Contenido Web (Login)	
Mantenimiento de roles (Crear, modificar, deshabilitar)	
Mantenimiento de usuario (Crear, modificar, deshabilitar)	
Mantenimiento de investigadores (Crear, modificar, deshabilitar)	
Mantenimiento de categorías de publicaciones (Líneas de investigación, Trabajos de Graduación, Trabajos de Posgrado,)	
Mantenimiento de publicaciones (Crear, editar, inhabilitar) asociadas a las categorías	Mantenimientos de fácil acceso e intuitivos para los usuarios
Mantenimiento de Testimonios (Crear, editar, inhabilitar)	
Mantenimientos de datos generales de la página web (Historia, Telefonos, Ubicaciones)	

Tabla Nº 9: Requerimientos Preliminares del Sistema

8.8 Historias de usuario

Las historias de usuarios que se realizaran fueron desarrolladas en conjunto con los usuarios involucrados en el proceso seleccionado para el desarrollo del proyecto. Para la estimación de los datos se tomó los siguientes criterios:

Prioridades en el desarrollo (PN): Se medirá en función al rango de: Alta, Media y baja, las cuales, serán asignadas por el Product Owner.

Importancia del Desarrollo (ID): Se asignará en un rango numérico del 1 al 100 entre el Product Owner y los miembros del equipo Scrum, donde:

- ✓ Todos los elementos con importancia ≥ 100 deben estar incluidos en el Sprint 1, por ser considerados de extrema importancia para el proyecto.
- ✓ Todos los elementos de importancia de 99-50 deberán estar incluidos en el Sprint 2, pero eso depende de la velocidad del Sprint.
- ✓ Los elementos con importancias de 49-25 los podremos incluir en el último Sprint, según el avance del equipo ya que son requisitos que no alteran el desarrollo de este o funcionalidades de este.

Tiempo Estimado (TS): Se asignará en un rango numérico del 1 al 20 entre el Product Owner y los miembros del equipo Scrum.

Tabla N° 10 - Historia de Usuario UH01

HISTORIA DE USUARIO	
ID: HU01	Usuario: Alvaro Romero
Nombre Historia: Acceso al Sistema (Login)	
Prioridad de desarrollo: Alta	Riesgo en desarrollo: Alta
Tiempo estimado: 25 horas.	Iteración asignada: 1
Programador Responsable: Alvaro Javier Romero Parrales.	
Descripción: Para el acceso al sistema se usará un usuario y una contraseña registrada en la base de datos del sistema, para poder tener acceso.	
Validación: <ul style="list-style-type: none">• Usuario previamente registrado y en estado activo.• 8 caracteres mínimo en la contraseña.	

Tabla N° 11 - Historia de Usuario UH02

HISTORIA DE USUARIO	
ID: HU02	Usuario: Álvaro Romero
Nombre Historia: Mantenimiento de Usuarios	
Prioridad de desarrollo: Alta	Riesgo en desarrollo: media
Tiempo estimado: 25 horas.	Iteración asignada: 1
Programador Responsable: Alvaro Javier Romero Parrales.	
Descripción: <ul style="list-style-type: none">• El usuario administrador podrá crear cuentas de usuarios con toda la información requerida como nombres, apellidos, teléfono, email y otros que puedan ser requeridos.• El usuario podrá editar y actualizar su propia información.• El usuario administrador podrá inhabilitar las cuentas de usuarios.	
Validación: <ul style="list-style-type: none">• Los usuarios solo podrán ser registrados una vez.• No todos los datos se podrán actualizar.	

Tabla N° 12 - Historia de Usuario UH03

HISTORIA DE USUARIO	
ID: HU03	Usuario: Álvaro Romero
Nombre Historia: Mantenimiento de Investigadores	
Prioridad de desarrollo: media	Riesgo en desarrollo: media

Tiempo estimado: 25 horas.	Iteración asignada: 1
Programador Responsable: Álvaro Javier Romero Parrales.	
Descripción: <ul style="list-style-type: none"> El usuario administrador podrá crear investigadores con toda la información requerida como título académico, departamento de trabajo, descripción de sus labores, fotos y otros que puedan ser requeridos El investigador podrá editar y actualizar su propia información. 	
Validación: <ul style="list-style-type: none"> El investigador solo podrá ser registrados una vez. El investigador tendrá que tener asociada una cuenta de usuario. 	

Tabla N° 13 - Historia de Usuario UH04

HISTORIA DE USUARIO	
ID: HU04	Usuario: Álvaro Romero
Nombre Historia: Mantenimiento de Categorías de Publicaciones	
Prioridad de desarrollo: media	Riesgo en desarrollo: media
Tiempo estimado: 15 horas.	Iteración asignada: 2
Programador Responsable: Álvaro Javier Romero Parrales.	
Descripción: <ul style="list-style-type: none"> El usuario administrador podrá crear categorías de publicaciones en las cuales los blogs y publicaciones realizadas por los investigadores se podrán agrupar. 	
Validación: <ul style="list-style-type: none"> Al inhabilitar una categoría esta seguirá siendo accesible desde el sitio web de los lectores, pero no podrán registrarse más publicaciones asociadas a esta. 	

Tabla N° 14 - Historia de Usuario UH05

HISTORIA DE USUARIO	
ID: HU05	Usuario: Álvaro Romero
Nombre Historia: Mantenimiento de Publicaciones	
Prioridad de desarrollo: media	Riesgo en desarrollo: media
Tiempo estimado: 80 horas.	Iteración asignada: 2
Programador Responsable: Álvaro Javier Romero Parrales.	
Descripción: <ul style="list-style-type: none">• Los usuarios podrán crear y actualizar publicaciones que reflejen los resultados de sus propias investigaciones y trabajos desarrollados.• Las publicaciones deberán almacenarse con toda la información requerida, título, fecha de publicación, autores, contenido, entre otros.	
Validación: <ul style="list-style-type: none">• Una publicación podrá ser creada con contenido multimedia.	

Tabla N° 15 - Historia de Usuario UH06

HISTORIA DE USUARIO	
ID: HU06	Usuario: Álvaro Romero
Nombre Historia: Mantenimiento de Testimonios	
Prioridad de desarrollo: baja	Riesgo en desarrollo: media
Tiempo estimado: 16 horas.	Iteración asignada: 3

Programador Responsable: Álvaro Javier Romero Parrales.
Descripción: <ul style="list-style-type: none"> Los investigadores y administradores podrán registrar testimonios de experiencias y/o vivencias de estudiantes o lectores relacionados al departamento de investigación o sus procesos. Se podrá registrar información de la persona que brinda el testimonio tales como nombres, correo, redes sociales entre otras.
Validación: <ul style="list-style-type: none"> El usuario podrá definir qué datos de la persona que brinda el testimonio podrá ser público.

Tabla N° 16 - Historia de Usuario UH07

HISTORIA DE USUARIO	
ID: HU07	Usuario: Álvaro Romero
Nombre Historia: Menú de Administración	
Prioridad de desarrollo: media	Riesgo en desarrollo: media
Tiempo estimado: 5 horas.	Iteración asignada: 3
Programador Responsable: Álvaro Javier Romero Parrales.	
Descripción: <ul style="list-style-type: none"> EL menú de administración deberá estar enlazado a todos los mantenimientos definidos. Acceso a todos los mantenimientos del sistema. 	
Validación: <ul style="list-style-type: none"> Deberá de comprobar el acceso a los usuarios a los mantenimientos según su rol. 	

Tabla N° 17 - Historia de Usuario UH08

HISTORIA DE USUARIO	
ID: HU08	Usuario: Álvaro Romero
Nombre Historia: Sitio Web PROITIC	
Prioridad de desarrollo: media	Riesgo en desarrollo: media
Tiempo estimado: 120 horas.	Iteración asignada: 4
Programador Responsable: Álvaro Javier Romero Parrales.	
Descripción: <ul style="list-style-type: none"> • Los lectores podrán acceder al sitio web donde podrán visualizar el resultado de los trabajos de los investigadores. • Podrán acceder a las publicaciones registradas, observar testimonios, revisar las categorías de publicaciones, información de los investigadores, información general del departamento de investigación, entre otras. 	
Validación: <ul style="list-style-type: none"> • Los clientes podrán mediante un formulario ponerse en contacto con el departamento de investigación. • El sitio web deberá ser amigable e intuitivo. 	

8.9 Lista de historias de usuario por orden de prioridad (BACKLOG)

Tabla N° 18 - Tareas por orden de prioridad

Historia de Usuario	Prioridad	Tiempo Estimado
---------------------	-----------	-----------------

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Acceso al Sistema (Login)	ALTA	25 horas
Mantenimiento de Usuarios	MEDIA	25 horas
Mantenimiento de Investigadores	MEDIA	25 horas
Mantenimiento de Categorías de Publicaciones	MEDIA	15 horas
Mantenimiento de Publicaciones	MEDIA	80 horas
Sitio Web PROITIC	MEDIA	120 horas
Menú de Administración	MEDIA	16 horas
Mantenimiento de Testimonios	BAJA	12 horas

8.10 Definición de Sprints

Se define la velocidad de desarrollo de cada Sprint según la importancia de las historias de usuario y el tiempo de trabajo del equipo de Scrum para el proyecto y la dedicación que se le dará al mismo.

El tiempo del equipo de trabajo está dado dentro de las jornadas laborales de 3 horas a la semana de lunes a sábado durante 3 meses, de los cuales, se obtiene como resultado la cantidad de horas de trabajo dedicados al proyecto por cada Sprint.

Tabla N° 19 - Días de trabajo dedicado del equipo por cada Sprint

Equipo SCRUM	Horas de trabajo por día	Horas de trabajo por Semana	Semanas de Trabajo	Total horas Laborables
Sergio Israel Polanco Pao	3 Horas	18 horas	12	216 horas

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Alvaro Javier Romero Parrales	3 Horas	18 horas	12	216 horas
Total de horas disponibles para el proyecto			432 horas	

Debido al tiempo de dedicación que se le dará al proyecto y las horas asignadas dentro de horario de trabajo se esperan tener algunas distracciones e impedimentos pero que están dentro de las estimaciones para el proyecto, por lo cual, el Product Owner da un factor de dedicación del 80% del tiempo comprendido para el mismo.

De acuerdo con el tiempo disponible y tomando en cuenta el nivel de importancia definido por cada historia de usuario se procede a agrupar las mismas y determinar la cantidad de Sprints para el proyecto, en donde se obtiene:

Sprint No 1		
Historia de Usuario	Prioridad	Tiempo Estimado
Acceso al Sistema (Login)	ALTA	25 horas
Mantenimiento de Usuarios	MEDIA	25 horas
Mantenimiento de Investigadores	MEDIA	25 horas
Total de horas del Sprint		75 horas

Tabla N° 20 – Estimación del Sprint N° 1

Sprint No 2		
Historia de Usuario	Prioridad	Tiempo Estimado
Mantenimiento de Categorías de Publicaciones	MEDIA	15 horas

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Mantenimiento de Publicaciones	MEDIA	80 horas
Total de horas del Sprint		95 horas

Tabla N° 21 – Estimación del Sprint N° 2

Sprint No 3		
Historia de Usuario	Prioridad	Tiempo Estimado
Menú de Administración	MEDIA	16 horas
Mantenimiento de Testimonios	BAJA	12 horas
Total de horas del Sprint		28 horas

Tabla N° 22 – Estimación del Sprint N° 3

Sprint No 4		
Historia de Usuario	Prioridad	Tiempo Estimado
Sitio Web PROITIC	MEDIA	120 horas
Total de horas del Sprint		120 horas

Tabla N° 23 – Estimación del Sprint N° 4

De acuerdo con la velocidad estimada de cada Sprint, el desarrollo del CMS se ejecutará en 4 Sprint, los mismos que han sido organizados por la importancia de

cada una de las historias de usuario y por el tiempo de duración de cada una de las mismas.

8.11 Planificación de los Sprints

Para el desarrollo de cada Sprint se han planificado revisiones y entregables para validar los avances obtenidos del desarrollo programado y así generar de manera retrospectiva las acciones de mejora para los siguientes desarrollos.

Por cada desarrollo de Sprint se mostrarán los avances a través del TaskBoard, donde se apreciarán las actividades en desarrollo, pendientes y finalizadas por cada historia de usuarios; además de mostrar el Burndown para ver la velocidad de desarrollo en la que se está dando el proyecto y determinar cuáles son las historias o actividades que están demandando mucho tiempo al desarrollo del proyecto o si las historias de usuario tiene pocas actividades de desarrollo y se están perdiendo recursos en ello.

Para validar la funcionalidad o conformidad de la elaboración de cada historia de usuario se realizarán pruebas de funcionalidad por cada historia de usuario y ver los aciertos y desaciertos de los mismos, los cuales, se verán reflejados en el informe de cierre del Sprint. Se procede a detallar la planificación de cada Sprint, indicando las fechas de revisión e historias de usuario comprendidas.

Tabla N° 24 – Planificación del Sprint No 1

SPRINT NUMERO 1	
Fecha de Inicio	18/09/2017
Fecha de Fin	09/10/2017
Revisión de los avances	Las revisiones se realizarán semanalmente. Las fechas de revisión serán las siguientes: ✓ 25/09/2017 ✓ 02/10/2017
Tareas a desarrollar	✓ Acceso al Sistema (Login) ✓ Mantenimiento de Usuarios ✓ Mantenimiento de Investigadores

--	--

Tabla N° 25 – Planificación del Sprint No 2

SPRINT NUMERO 2	
Fecha de Inicio	09/10/2017
Fecha de Fin	30/10/2017
Revisión de los avances	Las revisiones se realizarán semanalmente. Las fechas de revisión serán las siguientes: ✓ 16/10/2017 ✓ 23/10/2017 ✓ 30/10/2017
Tareas a desarrollar	✓ Mantenimiento de categorías de publicaciones ✓ Mantenimiento de publicaciones

Tabla N° 26 – Planificación del Sprint No 3

SPRINT NUMERO 3	
Fecha de Inicio	30/10/2017
Fecha de Fin	06/11/2017
Revisión de los avances	Las revisiones se realizarán semanalmente. Las fechas de revisión serán las siguientes: ✓ 06/11/2017
Tareas a desarrollar	✓ Menú de Administración ✓ Mantenimiento de Testimonios

Tabla N° 27 – Planificación del Sprint No 4

SPRINT NUMERO 4	
Fecha de Inicio	06/11/2017
Fecha de Fin	04/12/2017
Revisión de los avances	Las revisiones se realizarán semanalmente. Las fechas de revisión serán las siguientes: ✓ 13/11/2017 ✓ 20/11/2017 ✓ 27/11/2017 ✓ 04/12/2017
Tareas a desarrollar	✓ Sitio Web PROITIC

8.12 Desarrollo del Sistema

8.12.1 Sprint Numero 1

Semana 1

Se presenta el Taskboard de la Semana 1 en la cual se desarrolla el “Sprint 1”. Al finalizar dicha semana, las historias de usuario “Acceso al sistema”, “Mantenimiento de Usuarios” se encuentran en curso, en cambio la historia de usuario “Mantenimiento de Investigadores” se encuentra pendiente.

Tabla N° 28 – Desarrollo del Sprint 1, Semana 1

Inicio	18/09/2017
Fin	24/09/2017

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Nombre		Desarrollo del Sistema		
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)		✓	
	Mantenimiento de Usuarios		✓	
	Mantenimiento de Investigadores	✓		
Sprint No 2	Mantenimiento de Categorías de Publicaciones	✓		
	Mantenimiento de Publicaciones	✓		
Sprint No 3	Mantenimiento de Testimonios	✓		
	Menú de Administración	✓		
Sprint No 4	Sitio Web PROITIC	✓		

Semana 2

Se presenta el Taskboard de la Semana 2 en la cual se desarrolla el “Sprint 1”. Al finalizar dicha semana, la historia de usuario “Acceso al sistema” se encuentra finalizada, la historia de usuario “Mantenimiento de Usuarios” aún se encuentran en curso, en cambio la historia de usuario “Mantenimiento de Investigadores” inicia su desarrollo.

Tabla N° 29 – Desarrollo del Sprint 1, Semana 2

Inicio		25/09/2017		
Fin		01/09/2017		
Nombre		Desarrollo del Sistema		
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

	Mantenimiento de Usuarios		✓	
	Mantenimiento de Investigadores		✓	
Sprint No 2	Mantenimiento de Categorías de Publicaciones	✓		
	Mantenimiento de Publicaciones	✓		
Sprint No 3	Mantenimiento de Testimonios	✓		
	Menú de Administración	✓		
Sprint No 4	Sitio Web PROITIC	✓		

Semana 3

Se presenta el Taskboard de la Semana 3 en la cual se desarrolla el “Sprint 1”. Al finalizar dicha semana, las historias de usuario “Mantenimiento de Usuarios”, “Mantenimiento de Investigadores” son finalizadas.

Tabla N° 30 – Desarrollo del Sprint 1, Semana 3

Inicio	02/10/2017			
Fin	08/10/2017			
Nombre	Desarrollo del Sistema			
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones	✓		
	Mantenimiento de Publicaciones	✓		

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Sprint No 3	Mantenimiento de Testimonios	✓		
	Menú de Administración	✓		
Sprint No 4	Sitio Web PROITIC	✓		

Pruebas Funcionales

Se presentan los informes de la realización de las pruebas funcionales del Sprint Numero 01.

Tabla N° 31 – Informa de prueba funcional N° 01

PRUEBA FUNCIONAL				
Prueba No.	Prueba de Funcionalidad N° 01	Versión de ejecución	PF-AS01	
		Fecha de ejecución	02/10/2017	
Tarea	Acceso al Sistema			
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la validación de los campos, datos errados y los mensajes de respuesta que muestra.			
Caso de Prueba				
Precondiciones				
✓ Usuario creado para efectos de prueba en la base de datos				
Pasos de la prueba				
✓ Ingresar datos no válidos para validar campos ✓ Validar que el acceso funcione				
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema	
Campo	Valor			

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

[Usuario, Contraseña]	[Alvaromero, 12345678]	Acceso satisfactorio al sistema	Acceso satisfactorio al sistema
[Usuario, Contraseña]	[Alvaromero, 12345675]	Credenciales incorrectas	Usuario o contraseña incorrectos
Resultados de las pruebas			
Observaciones		Veredicto	
		<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo	

Tabla N° 32 – Informa de prueba funcional N° 02

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 02	Versión de ejecución	PF-MU01
		Fecha de ejecución	09/10/2017
Tarea	Mantenimiento de Usuarios		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la validación de los campos, datos errados, duplicidad de usuarios, editar usuario e inhabilitar usuarios.		
Caso de Prueba			
Precondiciones			
✓ Usuario administrador existente en la base de datos			
Pasos de la prueba			
✓ Validar los campos en el registro de usuarios ✓ Verificar que se puedan editar usuarios ya existentes ✓ Verificar que se pueda inhabilitar un usuario seleccionado			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Acción	Campos		

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Inserción	----	Los datos han sido ingresados	Los datos han sido ingresados satisfactoriamente
Inserción	----	El usuario ya se encuentra registrado	El usuario ya se encuentra registrado
Actualización	---	Los datos han sido actualizados	Los datos han sido actualizados satisfactoriamente
Actualización	----	Usuario Inhabilitado	Los datos han sido actualizados satisfactoriamente
Resultados de las pruebas			
Observaciones			Veredicto
			<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Tabla N° 33 – Informa de prueba funcional N° 03

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 03	Versión de ejecución	PF-MI01
		Fecha de ejecución	09/10/2017
Tarea	Mantenimiento de Investigadores		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la validación de los campos, datos errados, duplicidad de investigadores, editar datos e inhabilitar investigadores.		
Caso de Prueba			
Precondiciones			
✓ Usuarios registrados en la base de datos			
Pasos de la prueba			
✓ Validar los campos en el registro			
✓ Verificar que se puedan editar investigadores existentes			
✓ Verificar que se pueda inhabilitar un investigador seleccionado			

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Acción	Campos		
Inserción	----	El investigador ha sido registrado	Los datos han sido ingresados satisfactoriamente
Inserción	----	El investigador ya se encuentra registrado	El investigador ya se encuentra registrado
Actualización	----	Los datos han sido actualizados	Los datos han sido actualizados satisfactoriamente
Actualización	----	Investigador Inhabilitado	Los datos han sido actualizados satisfactoriamente
Resultados de las pruebas			
Observaciones			Veredicto
			<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Revisión del Sprint N° 1

Tabla N° 34 – Revisión del Sprint Numero 1

Nombre del Proyecto	Sistema de Manejador de Contenido Web para el Programa de Investigación en TIC (UNI-Nicaragua)
Lugar	Universidad Nacional de Ingeniería, Departamento de investigación
Fecha	09/10/2017
Numero de sprint	1
Personas convocadas a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao

Personas que asistieron a la reunión		Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
¿Qué salió bien en el Sprint?	¿Qué no salió bien en el Sprint?	Lecciones aprendidas	
La recuperación de datos entre los formularios de acceso, usuarios, administradores y la base de datos se ejecuta sin mayores complicaciones. El Taskboard aporta en gran medida a que el equipo de trabajo se mantenga al tanto de las actividades que se realizan, así como informar sobre el avance del proyecto.	El día de finalización del primer sprint fue el miércoles 06/10/2017, lo cual es un problema ya que son los lunes donde se realizan las reuniones con el Scrum Master.	Se sugiere siempre mantener actualizado el Taskboard para mantener informado al equipo y el mismo debe ser divulgado a todos los involucrados para no generar retrasos o se malinterpreten las necesidades y prioridades del desarrollo.	

8.12.2 Sprint Numero 2

Semana 4

Se presenta el Taskboard de la Semana 4 en la cual se desarrolla el “Sprint 2”. Al finalizar dicha semana, las historias de usuario “Mantenimiento de Categorías de Publicaciones”, “Mantenimiento de Publicaciones” se encuentran en curso.

Tabla N° 35 – Desarrollo del Sprint 2, Semana 4

Inicio	09/10/2017
Fin	15/10/2017
Nombre	Desarrollo del Sistema

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones		✓	
	Mantenimiento de Publicaciones		✓	
Sprint No 3	Mantenimiento de Testimonios	✓		
	Menú de Administración	✓		
Sprint No 4	Sitio Web PROITIC	✓		

Semana 5

Se presenta el Taskboard de la Semana 5 en la cual se desarrolla el “Sprint 2”. Al finalizar dicha semana, la historia de usuario “Mantenimiento de Categorías de Publicaciones” se encuentra finalizada, “Mantenimiento de Publicaciones” sigue en curso.

Tabla N° 36 – Desarrollo del Sprint 2, Semana 5

Inicio		16/10/2017		
Fin		22/10/2017		
Nombre		Desarrollo del Sistema		
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones		✓	
Sprint No 3	Mantenimiento de Testimonios	✓		
	Menú de Administración	✓		
Sprint No 4	Sitio Web PROITIC	✓		

Semana 6

Se presenta el Taskboard de la Semana 6 en la cual se desarrolla el “Sprint 2”. Al finalizar dicha semana, la historia de usuario “Mantenimiento de Publicaciones” aún se encuentra en curso

Tabla N° 37 – Desarrollo del Sprint 2, Semana 6

Inicio		23/10/2017		
Fin		29/10/2017		
Nombre		Desarrollo del Sistema		
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones			✓
Sprint No 3	Mantenimiento de Testimonios	✓		
	Menú de Administración	✓		

Sprint No 4	Sitio Web PROITIC	✓		
-------------	-------------------	---	--	--

Pruebas Funcionales

Se presentan los informes de la realización de las pruebas funcionales del Sprint Numero 02.

Tabla N° 38 – Informa de prueba funcional N° 04

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 04	Versión de ejecución	PF-MC01
		Fecha de ejecución	23/10/2017
Tarea	Mantenimiento de Categorías de Publicaciones		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la validación de los campos, datos errados y los mensajes de respuesta que muestra.		
Caso de Prueba			
Precondiciones			
✓ No aplica			
Pasos de la prueba			
✓ Ingresar datos no válidos para validar campos ✓ Inserción y modificación de categorías.			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Acción	Campos		
Inserción	----	Categoría creada satisfactoriamente	Los datos han sido ingresados satisfactoriamente
Actualización	---	Categoría modificada satisfactoriamente	Los datos han sido actualizados satisfactoriamente
Resultados de las pruebas			

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Observaciones	Veredicto
	<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Tabla N° 39 – Informa de prueba funcional N° 05

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 05	Versión de ejecución	PF-MP01
		Fecha de ejecución	23/10/2017
Tarea	Mantenimiento de Publicaciones		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la validación de los campos, datos errados, duplicidad de usuarios, editar usuario e inhabilitar usuarios.		
Caso de Prueba			
Precondiciones			
✓ Categorías de publicaciones existentes			
Pasos de la prueba			
✓ Validar los campos en la creación de publicaciones ✓ Verificar que se puedan editar las publicaciones ✓ Verificar que solo los investigadores puedan crear publicaciones			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Acción	Campos		
Inserción	----	Publicación creada	La publicación fue creada satisfactoriamente
Actualización	----	Publicación modificada	La publicación fue modificada satisfactoriamente
Actualización	----	Publicación Inhabilitado	La publicación fue modificada satisfactoriamente

Resultados de las pruebas	
Observaciones	Veredicto
No se permite la carga de multimedia	<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Tabla N° 40 – Informa de prueba funcional N° 06

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 06	Versión de ejecución	PF-MP02
		Fecha de ejecución	30/10/2017
Tarea	Mantenimiento de Publicaciones		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la carga de archivos multimedia		
Caso de Prueba			
Precondiciones			
✓ Categorías de publicaciones existentes			
Pasos de la prueba			
✓ Validar que se permita adjuntar archivos multimedia a una publicación			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Acción	Campos		
Adjuntar imagen	----	Publicación creada	La publicación fue creada satisfactoriamente
Adjuntar archivo	----	Publicación modificada	La publicación fue modificada satisfactoriamente

Resultados de las pruebas	
Observaciones	Veredicto
	<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Revisión del Sprint N° 2

Tabla N° 41 – Revisión del Sprint Numero 2

Nombre del Proyecto	Sistema de Manejador de Contenido Web para el Programa de Investigación en TIC (UNI-Nicaragua)	
Lugar	Universidad Nacional de Ingeniería, Departamento de investigación	
Fecha	30/10/2017	
Numero de sprint	2	
Personas convocadas a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
Personas que asistieron a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
¿Qué salió bien en el Sprint?	¿Qué no salió bien en el Sprint?	Lecciones aprendidas
Al realizar cada módulo del sistema de manera independiente, siguiendo el orden planteado, no hubo problemas al momento de integrarlos entre ellos.	Se encontraron inconvenientes con la carga de archivos multimedia, ya que se debía de tomar la decisión de almacenar los archivos en la BD o en carpetas físicas. Este dilema retraso el desarrollo.	Se recomienda seguir con la programación por módulos, agrupados por actividades independientes, ya que facilita su elaboración y ayuda a mantener al equipo dentro del mismo enfoque de desarrollo.

Los tiempos estimados para el desarrollo de cada actividad del sprint fueron los necesarios para terminar cada una de las tareas dentro.		
--	--	--

8.12.3 Sprint Numero 3

Semana 7

Se presenta el Taskboard de la Semana 7 en la cual se desarrolla el “Sprint 3”. En esta semana se inician y finalizan las historias de usuarios “Mantenimiento de Testimonios” y “Menú Administrador”.

Tabla N° 42 – Desarrollo del Sprint 3, Semana 7

Inicio	30/10/2017			
Fin	06/11/2017			
Nombre	Desarrollo del Sistema			
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones			✓
Sprint No 3	Mantenimiento de Testimonios			✓

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

	Menú de Administración			✓
Sprint No 4	Sitio Web PROITIC	✓		

Pruebas Funcionales

Se presentan los informes de la realización de las pruebas funcionales del Sprint Numero 03.

Tabla N° 43 – Informa de prueba funcional N° 07

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 07	Versión de ejecución	PF-MT01
		Fecha de ejecución	06/11/2017
Tarea	Mantenimiento de Testimonios		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la validación de los campos, datos errados y los mensajes de respuesta que muestra.		
Caso de Prueba			
Precondiciones			
✓ No aplica			
Pasos de la prueba			
✓ Ingresar datos no válidos para validar campos ✓ Inserción y modificación de testimonios.			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Acción	Campos		
Inserción	----	Categoría creada satisfactoriamente	Categoría creada satisfactoriamente
Actualización	---	Categoría modificada satisfactoriamente	Los datos han sido actualizados satisfactoriamente

Resultados de las pruebas	
Observaciones	Veredicto
	<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Tabla N° 44 – Informa de prueba funcional N° 08

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 08	Versión de ejecución	PF-MA01
		Fecha de ejecución	06/11/2017
Tarea	Menú de Administración		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a los roles de usuarios.		
Caso de Prueba			
Precondiciones			
✓ Usuarios registrados			
Pasos de la prueba			
✓ Acceder a las distintas opciones del sistema desde el menú de administración			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Usuario			
Investigador		Acceso a todas las opciones del sistema	Acceso denegado en ciertas opciones de administración
Administrador		Acceso a todas las opciones del sistema	Acceso permitido
Resultados de las pruebas			

Observaciones	Veredicto
	<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Revisión del Sprint N° 3

Tabla N° 45 – Revisión del Sprint Numero 3

Nombre del Proyecto	Sistema de Manejador de Contenido Web para el Programa de Investigación en TIC (UNI-Nicaragua)	
Lugar	Universidad Nacional de Ingeniería, Departamento de investigación	
Fecha	06/11/2017	
Numero de sprint	3	
Personas convocadas a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
Personas que asistieron a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
¿Qué salió bien en el Sprint?	¿Qué no salió bien en el Sprint?	Lecciones aprendidas
Los tiempos estimados para el desarrollo de cada actividad del sprint fueron los necesarios para terminar cada una de las tareas dentro.		Se recomienda hacer un análisis de todas las actividades que se puedan presentar dentro del desarrollo del proyecto para que las fechas y plazos dados no se prolonguen demasiado.

8.12.4 Sprint Numero 4

Semana 8

Se presenta el Taskboard de la Semana 8 en la cual se desarrolla el “Sprint 4”. En esta semana se inicia la historia de usuario “Sitio Web PROITIC”.

Tabla N° 45 – Informa de prueba funcional N° 08

Inicio	06/11/2017			
Fin	12/11/2017			
Nombre	Desarrollo del Sistema			
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones			✓
Sprint No 3	Mantenimiento de Testimonios			✓
	Menú Administrador			✓
Sprint No 4	Sitio Web PROITIC		✓	

Semana 9

Se presenta el Taskboard de la Semana 9 en la cual se desarrolla el “Sprint 4”. En esta semana la historia de usuario “Sitio Web PROITIC” aún sigue en curso.

Tabla N° 46 – Informa de prueba funcional N° 09

Inicio	13/11/2017
---------------	------------

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Fin	19/11/2017			
Nombre	Desarrollo del Sistema			
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones			✓
Sprint No 3	Mantenimiento de Testimonios			✓
	Menú Administrador			✓
Sprint No 4	Sitio Web PROITIC		✓	

Semana 10

Se presenta el Taskboard de la Semana 10 en la cual se desarrolla el “Sprint 4”. En esta semana la historia de usuario “Sitio Web PROITIC” aún sigue en curso.

Tabla Nº 47 – Informa de prueba funcional Nº 10

Inicio	20/11/2017			
Fin	26/11/2017			
Nombre	Desarrollo del Sistema			
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones			✓
Sprint No 3	Mantenimiento de Testimonios			✓
	Menú Administrador			✓
Sprint No 4	Sitio Web PROITIC		✓	

Semana 11

Se presenta el Taskboard de la Semana 11 en la cual se desarrolla el “Sprint 4”. En esta semana la historia de usuario “Sitio Web PROITIC” es finalizada.

Tabla N° 48 – Informa de prueba funcional N° 11

Inicio	27/11/2017			
Fin	03/12/2017			
Nombre	Desarrollo del Sistema			
Sprint	Historia de Usuario	Pendiente	En Curso	Hecho
Sprint No 1	Acceso al Sistema (Login)			✓
	Mantenimiento de Usuarios			✓
	Mantenimiento de Investigadores			✓
Sprint No 2	Mantenimiento de Categorías de Publicaciones			✓
	Mantenimiento de Publicaciones			✓
Sprint No 3	Mantenimiento de Testimonios			✓
	Menú Administrador			✓

Sprint No 4	Sitio Web PROITIC			✓
-------------	-------------------	--	--	---

Pruebas Funcionales

Se presentan los informes de la realización de las pruebas funcionales del Sprint Numero 04.

Tabla N° 49 – Informa de prueba funcional N° 08

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 08	Versión de ejecución	PF-SW01
		Fecha de ejecución	19/11/2017
Tarea	Sitio Web PROITIC		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la presentación de la información publicada		
Caso de Prueba			
Precondiciones			
<div>✓ Investigadores registrados</div> <div>✓ Categorías de publicaciones registradas</div>			
Pasos de la prueba			
<div>✓ Información de los investigadores</div> <div>✓ Categorías en el menú</div> <div>✓ Datos Generales</div>			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Modulo			
Menú		El menú presenta las distintas categorías creadas	El menú presenta las distintas categorías creadas

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Investigadores	Se presenta la información registrada de los investigadores	Se presenta la información registrada de los investigadores
Datos Generales	Se presenta la información registrada acerca del departamento	Se presenta la información registrada acerca del departamento
Resultados de las pruebas		
Observaciones		Veredicto
		<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Tabla N° 50 – Informa de prueba funcional N° 09

PRUEBA FUNCIONAL			
Prueba No.	Prueba de Funcionalidad N° 08	Versión de ejecución	PF-SW01
		Fecha de ejecución	03/12/2017
Tarea	Sitio Web PROITIC		
Descripción del caso de prueba:	Se procederá a realizar pruebas con respecto a la presentación de la información publicada		
Caso de Prueba			
Precondiciones			
✓ Publicaciones registradas ✓ Testimonios registrados			
Pasos de la prueba			
✓ Visualización de publicaciones ✓ Visualización de testimonios			
Datos de Entrada		Respuesta/Acción esperada	Respuesta/Acción del sistema
Modulo			

**Sistema de Manejador de Contenido Web para Programa de Investigación en TIC
(UNI-NICARAGUA)**

Publicaciones	Se presenta y se puede acceder a todas las publicaciones activas.	Se presenta y se puede acceder a todas las publicaciones activas.
Investigadores	Se presenta la información de todos los testimonios registrados	Se presenta la información de todos los testimonios registrados
Resultados de las pruebas		
Observaciones		Veredicto
		<input checked="" type="checkbox"/> Paso <input type="checkbox"/> Fallo

Revisión del Sprint N° 4

Tabla N° 51 – Revisión del Sprint Numero 4

Nombre del Proyecto	Sistema de Manejador de Contenido Web para el Programa de Investigación en TIC (UNI-Nicaragua)	
Lugar	Universidad Nacional de Ingeniería, Departamento de investigación	
Fecha	03/12/2017	
Numero de sprint	4	
Personas convocadas a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
Personas que asistieron a la reunión	Johnny Flores Delgadillo Alvaro Javier Romero Parrales Sergio Israel Polanco Pao	
¿Qué salió bien en el Sprint?	¿Qué no salió bien en el Sprint?	Lecciones aprendidas
Al tener ya todos los módulos creados y bien indexados la creación	Los tiempos de desarrollo para la historia de usuario fueron muy cambiantes debido a que las actividades estuvieron	Se recomienda hacer un análisis de todas las actividades que se puedan presentar dentro del desarrollo del proyecto para que

del sitio web fue más sencilla.	sujetas a distracciones no planeadas.	las fechas y plazos dados no se prolonguen demasiado.
---------------------------------	---------------------------------------	---

8.13 Liberación del Sistema

Se constató la funcionalidad y la calidad implementada en el software partiendo de la documentación de los requerimientos de la aplicación, de esta forma se le entregó al departamento de investigación y tecnología la versión final de la aplicación “Sistema Manejador de Contenido Web”, y se culminó la documentación de la monografía.

Capítulo 9: Conclusiones

La realización de este proyecto permitió elaborar un Sistema de Manejador de Contenido Web para el Programa de Investigación en TIC (UNI-Nicaragua).

A fin de que el desarrollo de la aplicación se realizara en forma precisa y metodológica se planteó el objetivo general y los objetivos específicos, los cuales, cómo se refleja en el contenido de este documento, se cumplieron en su totalidad. En el desarrollo del proyecto se usó la metodología ágil SCRUM, ejecutándose las fases establecidas del mismo, para la planificación, recolección de datos y demás etapas requeridas para el exitoso cumplimiento de los objetivos propuestos.

Los objetivos específicos de la monografía se cumplieron de la siguiente manera, el objetivo 1 se cumplió a lo largo del proyecto en la ejecución de: “Historias de Usuarios.”, los objetivos específicos 2,3,4,5 se efectuó en los 4 Sprint desarrollados a lo largo del proyecto y se lograron cumplir en su totalidad en la finalización del último del sprint.

Una vez culminados todos los puntos propuestos para el desarrollo del proyecto y habiendo cumplido con los cinco objetivos específicos de la monografía, se llega a la conclusión que fue factible lograr su elaboración, habiendo culminado con éxito el “Sistema de Manejador de Contenido Web para el Programa de Investigación en TIC (UNI-Nicaragua)”.

Capítulo 10: Bibliografía

- [1] GILBANE, F. "Whats content management?". The Gilbane Report, vol. 8, n. 8, 2000
(http://www.gilbane.com/gilbane_report.pl/6/What_is_Content_Management)
- [2] Django Software Foundation. [Online]. Available: (<https://docs.djangoproject.com/en/1.10/>)
- [3] Free Software Foundation, Inc. [Online]. Available: (<https://www.gnu.org/philosophy/free-sw.es.html/>)
- [4] MURAS: APROXIMACION A LOS CMS. <http://mosaic.uoc.edu/2009/12/23/unaaproximacion-a-los-gestores-de-contenidos-cms/>, 2011
- [5] Wikipedia Enciclopedia. CMS. <http://es.wikipedia.org/wiki/CMS>, Abril 2011
- [6] Python Foundation. Available: (<https://www.python.org/about/>)
- [7] Bootstrap. Available: (<https://getbootstrap.com/docs/3.3/about/>)
- [8] JavaScript. Available: (<https://www.javascript.com/about>)
- [9] Javier J. Gutierrez. "¿Qué es un framework web?".
(http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf)
- [10] Sergio Luján Mora. "Programación de Aplicaciones Web". Editorial Club Universitario, 2002
- [11] Wikipedia Enciclopedia. Framework. <https://es.wikipedia.org/wiki/Framework>, Enero 2013
- [12] jQuery. [Online]. Available: (<https://jquery.com/>)
- [13] Bootstrap. [Online]. Available: (<http://getbootstrap.com/>)
- [14] Damián Pérez Valdés. (julio 3 2007). ¿Qué es Javascript?. julio 3 2007, de Maestros del Web
Sitio web: <http://www.maestrosdelweb.com/que-es-javascript/>
- [15] Miguel Angel Alvarez. (Noviembre 19 2003). Qué es Python. Noviembre 19 2003, de
DesarrolloWeb.com Sitio web: <https://desarrolloweb.com/articulos/1325.php>
- [16] Wikipedia Enciclopedia. Javascript. <https://es.wikipedia.org/wiki/JavaScript>
- [17] María. (agosto 3 2016). Qué es Bootstrap y cuáles son sus ventajas. Available:
<https://puntoabierto.net/blog/que-es-bootstrap-y-cuales-son-sus-ventajas>
- [18] Joaquín Barberá (septiembre 2 2014). Clasificación y características de los CMS o Gestores
de contenidos: <http://www.xn--diseowebmurcia1-1qb.es/clasificacion-caracteristicas-cms-gestores-de-contenidos/>
- [19] Richard Stallman (agosto 2015). Free Software Foundation Staff. «What is free software and
why is it so important for society?» : <https://www.fsf.org/about/what-is-free-software>

[20] Proyectos Ágiles. (septiembre,2017). Scrum. Barcelona: Proyectos Agiles org. Recuperado de <https://proyectosagiles.org/>