OCTUBRE, 2016



**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**ASIGNATURA:**

PROYECTO INTEGRADOR DE INGENIERÍA DE SOFTWARE

11:00-12:00 P.M.

**MAESTRA:**

ARCELIA JUDITH BUSTILLOS MARTINEZ

**ALUMNOS:**

RAMIREZ SALAZAR LUIS FERNANDO

JOSÉ MARIO SANZ LÓPEZ.

**ARTEFACTOS DE LA FASE DE INICIO**

**TABLA DE CONTENIDO**

1. VISIÓN Y ANÁLISIS DE NEGOCIO.
   1. Visión.
   2. Análisis de negocio.
   3. Historia de revisiones.
   4. Introducción.
   5. Enunciado del problema.
   6. Enunciado de la posición en el mercado del producto.
   7. Descripción del personal involucrado.
   8. Objetivos de alto nivel y problemas claves del personal involucrado.
   9. Objetivos a nivel de usuario.
   10. Perspectiva del producto.
   11. Resumen de beneficiarios
   12. Resumen de las características del sistema
   13. Otros requisitos y restricciones.
2. MODELO DE CASOS DE USO.
   1. Historia de revisiones
   2. Requisitos Funcionales y No Funcionales
   3. Lista Actor-Semántica.
   4. Lista Actor-Objetivo.
   5. Lista de casos de uso.
   6. Descripción breve de casos de uso.
   7. Descripción completa de casos de uso.
   8. Diagrama de casos de uso.
3. ESPECIFICACIONES COMPLEMENTARIA.
   1. Historia de revisiones
   2. Introducción
   3. Funcionalidad
      1. Registro y gestión de errores
      2. Reglas de negocio conectables
      3. Seguridad
   4. Facilidad de uso
      1. Factores Humanos
   5. Fiabilidad
      1. Capacidad de recuperación
   6. Rendimiento
   7. Interfaces
      1. Interfaces y hardware destacable
      2. Interfaces software
   8. Reglas del dominio (negocio).
4. GLOSARIO.
   1. Historia de revisiones
   2. Definiciones
5. LISTA DE RIESGOS Y PLAN DE GESTIÓN DE RIESGOS.
   1. Lista de riesgos
   2. Plan de gestión de riegos
      1. Metodología de gestión de riegos
      2. Roles y responsabilidades de la gestión de riesgos
      3. Calendario de la gestión de riesgos
      4. Matriz de probabilidad e impacto
      5. Tabla de análisis de riesgos
6. PROTOTIPOS Y PRUEBAS DE CONCEPTOS.
   1. Prototipos
   2. Pruebas de conceptos
7. PLAN DE ITERACIÓN.
8. FASE PLAN Y PLAN DE DESARROLLO DE SOFTWARE.
   1. Plan de desarrollo del software
9. MARCO DE DESARROLLO.
10. **VISION Y ANALISIS DEL NEGOCIO**
    1. VISION

Satisfacer las necesidades de los clientes con los artículos, para asegurar una relación con nuestros clientes, obteniendo de esta manera una adecuada rentabilidad y garantizando así nuestra permanencia y el crecimiento de nuestra empresa.

* 1. ANALISIS DEL NEGOCIO
* Perspectivas del sector
* Visión del servicio
* Puntos fuertes y ventajas
* Proveedores
* Clientes
  1. HISTORIA DE REVISIONES

|  |  |  |  |
| --- | --- | --- | --- |
| VERSION | FECHA | DESCRIPCION | AUTOR |
| 1ra Versión | 20/Octubre/2016 | Primera versión del borrador del proyecto. | Luis Ramírez |
| 2da Versión | 23/Octubre/2016 | Segunda versión, se agregó más información a esta fase del inicio. | Mario Sanz |
| 3ra Revisión | 27/Octubre/2016 | Tercera versión, refinamiento en la fase para poder avanzar con el proyecto. | Mario Sanz |

* 1. INTRODUCCIÓN:

TecShop es una de las muchas empresas de negocio convencional, ósea una tienda en línea las cuales ofrezcan artículos a sus clientes.

Por otra parte, el personal autorizado podrá realizar acciones como el modificado de los pedidos realizados, la inserción de nuevos artículos a la base de datos o la corrección de datos erróneos que puedan aparecer en el registro de algún cliente.

* 1. ENUNCIADO DEL PROBLEMA:

“TecShop frente a la innovación y creatividad”

En un mundo caracterizado por los cambios rápidos, a las empresas se les plantea cada vez más problemas más difíciles que no solo afectan a su organización sino que también tienen impacto en la sociedad. Por eso, cada vez tendrán mayor necesidad de enfrentar las situaciones de un modo creativo e innovador.

* 1. ENUNCIADO DE LA POSICION EN EL MERCADO DEL PRODUCTO:

Ser original en esta parte es esencial, por lo que el mercado de nuestra aplicación, será dirigido a todas las personas que les encanta lo relacionado con las tecnología, ósea lo más nuevo en el mundo. Con el desarrollo de esta aplicación la gente podrá ser parte de una experiencia en la que los involucramos a ser parte de nuestra empresa con comentarios o sugerencias, con la finalidad de mejorar la aplicación.

* 1. DESCRIPCIÓN DEL PERSONAL INVOLUCRADO:

Administrador:

* Se llamara administrador a toda aquella persona que tenga acceso autorizado para poder administrar la aplicación.

Usuario/Cliente:

* Se llamara usuario o cliente a cualquier persona que tenga acceso a la aplicación, ya sea como visitante o como cliente registrado en la BD.
  1. DESCRIPCION DE ALTO NIVEL Y PROBLEMAS CLAVES DEL PERSONAL INVOLUCRADO.

|  |  |  |  |
| --- | --- | --- | --- |
| OBJETIVO DE ALTO NIVEL | PRIORIDAD | PROBLEMAS E INQUIETUDES | SOLUCIONES ACTUALES |
| Realizar un procesamiento rápido e integrado | Alta | Se reduce la velocidad cuando se aumenta el tráfico de usuarios en la aplicación. | Los dispositivos utilizados para el manejo de las consultas o peticiones de los clientes son muy rápidos y logran atender eficientemente. |
| Selección de articulo correcto | Alta | El usuario selecciona el artículo que desea sin saber que no está disponible | Se implementara un aviso, para que el cliente pueda observar cuando un artículo este de faltante en el stock de la tienda. |
| Seguridad | Alta | El usuario quiere tener total seguridad de los datos que proporciono, para que no se generen errores en varios casos en los que se le involucra. | Se tiene una medida en cuanto a esos tipos de acciones, protegemos sus datos con buena seguridad y respaldando sus compras con diferentes tipos de servicios. |

* 1. OBJETIVOS A NIVEL DEL USUARIO

Cada actor tiene un papel para cada caso de uso en el cual interviene.

* El usuario invitado no se ha identificado en el sistema, por lo que solo puede hacer consultas, es la parte pública y simular una compra.

Sus funciones son:

* Alta usuario
* Login
* Consultas catálogo
* Añadir, eliminar y simular compra
* Usuario registrado o cliente, está identificado en el sistema.

Sus funciones son:

* Confirmar compra
* Consulta, modificación y baja usuario
* Ver pedidos pendientes
* Administrador, se encargara del mantenimiento de la tienda. Tiene acceso a toda la parte privada de la aplicación.

Sus funciones son:

* Alta, baja, modificar artículos.
* Alta, baja, modificar catálogos.
* Otras tareas especiales.
  1. PERSPECTIVA DEL PRODUCTO

TecShop va a consistir en una interfaz web dinámica que permita por una parte a los clientes que estén registrados o que la visiten consultar los artículos que estén disponibles, así como todos los datos sobre los artículos, para de esta manera los usuarios puedan mantenerse informados de los artículos en la tienda.



* 1. RESUMEN DE BENEFICIARIOS

|  |  |
| --- | --- |
| CARACTERISTICA SOPORTADA | BENEFICIO DEL PERSONAL INVOLUCRADO |
| Funcionalmente, el sistema proporcionara todos los servicios tipos que requiere la organización para brindar la mejor calidad al cliente. | Servicios de venta eficientes y automáticos. |
| Los usuarios podrán comentar y calificar cualquier artículo. | Los administradores estarán al tanto de lo que debe de mejorar en su servicio. |

* 1. RESUMEN DE LAS CARACTERISTICAS DEL SISTEMA
* Tendrá una sección de registrar usuarios
* Tendrá una sección para acceder como usuario
* Tendrá una sección solo accesible para los administradores del sistema
* Tendrá un módulo para configurar las categorías
* Tendrá un módulo para configurar los artículos
* Si el usuario está activo podrá dejar comentarios y calificar el producto.
* Será posible realizar pedidos individuales
* Cada categoría mostrara una lista de todos los productos de dicha categoría
* Cada artículo tendrá su propia página de la cual se podrán ver sus detalles
  1. OTROS REQUISITOS Y RESTRICCIONES

ALGUNAS DE LAS RESTRICCIONES:

* Debe de ser una aplicación web, utilizando html5.
* No debe de contener enlaces rotos dentro de la aplicación.

Abarca las restricciones de diseño, fiabilidad, rendimiento, etcétera. Diríjase a la Especificación Complementaria y a los casos de uso.

1. **MODELO DE CASOS DE USO.**
   1. HISTORIA DE REVISIONES

|  |  |  |  |
| --- | --- | --- | --- |
| VERSION | FECHA | DESCRIPCION | AUTOR |
| 1ra versión | 20/Octubre/2016 | Primera versión con la fase de inicio modelo de casos de uso. | Luis Ramírez |
| 2da versión | 22/Octubre/2016 | Segunda versión con un manejo información y retroalimentación de los casos de uso. | Mario Sanz |
| 3ra versión | 27/Octubre/2016 | Refinamiento de los casos de uso, para tener completa esta fase de inicio. | Mario Sanz |

* 1. REQUISITOS FUNCIONALES Y NO FUNCIONALES:

REQUISITOS FUNCIONALES:

* Catálogo de artículos organizados:

La aplicación debe permitir al cliente poder navegar por los diferentes catálogos viendo los artículos que hay en cada una de ellas.

* Carrito de compra.

Un cliente podrá seleccionar unos artículos y añadirlos al carrito de compra. Este carrito contará con todas las funciones típicas (añadir elemento, borrar).

* Área privada: Inserción, modificación y eliminación de artículos.

La aplicación implementará un área privada donde los administradores de la aplicación contarán con una herramienta que servirá para añadir nuevos artículos y categorías, modificar características de los ya existentes, o eliminar artículos y categorías.

REQUISITOS NO FUNCIONALES:

* Se pide específicamente que la aplicación sea rápida en cuanto a las peticiones de los usuarios.
* Se pide que la aplicación sea segura en cuanto a los datos e integridad del usuario.
* Que este desarrollado con el framework Laravel 5.3.
* Etc.
  1. LISTA ACTOR-SEMÁNTICA.

El usuario invitado:  
Es el tipo de usuario que no se ha identificado en el sistema, por lo que solo puede hacer consultas, es la parte pública y simular una compra.

Usuario registrado o cliente:  
Es el usuario que está identificado en el sistema.

Administrador:  
se encargara del mantenimiento de la tienda. Tiene acceso a toda la parte privada de la aplicación.

* 1. LISTA ACTOR-OBJETIVO.

|  |  |
| --- | --- |
| ACTOR | OBJETIVO |
| USUARIO | * Alta usuario * Login * Consultas catálogo * Añadir, eliminar y simular compra * Confirmar compra * Consulta, modificación y baja usuario * Ver pedidos pendientes |
| ADMINISTRADOR | * Alta, baja, modificar artículos. * Alta, baja, modificar catálogos. * Otras tareas especiales. |

* 1. LISTA DE CASOS DE USO.
* Alta de usuario
* Baja de usuario
* Consulta de usuario
* Modifica el usuario
* Registrarse
* Alta de Catálogo
* Baja de Catálogo
* Consulta de Catálogo
* Modifica el Catálogo
* Alta de articulo a catálogo
* Baja de artículo a catálogo
* Modifica artículo en catálogo
* Añadir artículo al carrito
* Elimina artículo al carrito
* Consulta artículos del carrito
  1. DESCRIPCIÓN BREVE DE LOS CASOS DE USO.
* Alta de usuario: Da de alta un usuario en la base de datos, se mira que no exista en la BD.
* Baja de usuario: Da de baja un usuario en la BD.
* Consulta de usuario: Muestra los datos del usuario registrados en la BD.
* Modifica el usuario: Modifica los datos del usuario registrados en la BD.
* Registrarse: verifica que el usuario y contraseña existan en la BD.
* Alta de Catálogo: Dar de alta un catálogo en la BD.
* Baja de Catálogo: Dar de baja un catálogo en la BD.
* Consulta de Catálogo: Lista todos los artículos correspondientes a un catálogo
* Modifica el Catálogo: Modifica los datos del catálogo registrado en la BD.
* Alta de artículo a catálogo: Dar de alta un artículo en un catálogo en la BD.
* Baja de artículo a catálogo: Dar de baja un artículo en un catálogo en la BD.
* Modifica artículo en catálogo: Modificar un artículo en un catálogo en la BD.
* Añadir artículo al carrito: Agrega un artículo al carrito del usuario.
* Elimina artículo al carrito: Elimina un artículo al carrito del usuario.
* Consulta artículo del carrito: Consulta un artículo del carrito del usuario.
  1. DESCRIPCIÓN COMPLETA DE LOS CASOS DE USO. (10% al 20%)

1. **ALTA DE USUARIO:**

Resumen de la funcionalidad:  
Da de alta un usuario en la base de datos, se mira que no exista en la BD, introducir los datos del usuario.

Actor: Usuario invitado  
Casos de usos relacionados: Registrarse, consulta usuario.  
Precondición: El usuario no puede existir en la base de datos.

Función principal:

* El usuario escoge la opción de Alta usuario
* El sistema le presenta un formulario a rellenar por el usuario donde le pide: Login, password, repetir password, nombre, apellidos, dirección, código postal, población, provincia, teléfono, email, cuenta, etc.

1. **BAJA DE USUARIO:**

Resumen de la funcionalidad:  
Modifica los datos del usuario registrados en la base de datos.

Actor: Usuario registrado   
Casos de usos relacionados: Baja usuario, Alta usuario, Modifica usuario, Registrarse.  
Precondición: El usuario debe de existir en la base de datos

Función principal:

* El usuario pulsa la opción consulta datos.
* El sistema mostrara los datos del usuario.

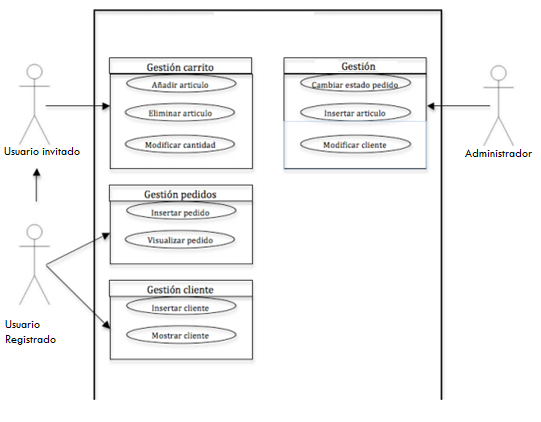
1. **MODIFICA EL USUARIO:**

Resumen de la funcionalidad:   
Modifica los datos del usuario registrados en la base de datos.

Actor: Usuario identificado  
Casos de usos relacionados: Consulta usuario.  
Precondición: El usuario se ha de registrar y existe en la base de datos.

Función principal:

* El usuario está consultando sus datos, edita los campos del formulario deseado, y pulsa la opción modificar datos.
* El sistema grabara los datos del usuario.
  1. DIAGRAMA DE CASOS DE USO.



1. **ESPECIFICACIÓN COMPLEMENTARIA.**
   1. HISTORIA DE REVISIONES.

|  |  |  |  |
| --- | --- | --- | --- |
| VERSION | FECHA | DESCRIPCION | AUTOR |
| 1ra versión | 21/Octubre/2016 | Primera versión con de las especificaciones complementarias con poco refinamiento | Luis Ramírez |
| 2da versión | 22/Octubre/2016 | Segunda versión con un manejo mayor de información. | Mario Sanz |
| 3ra versión | 27/Octubre/2016 | Refinamiento de las especificaciones complementarias para tener completa esta fase de inicio. | Mario Sanz |

* 1. FUNCIONALIDAD
     1. REGISTRO Y GESTIÓN DE ERRORES

Registrar un control adecuado de los percances o errores que se van presentando para el manejo correctamente de la aplicación, ya que en algún otro momento se pueden presentar.

* + 1. REGLAS DEL NEGOCIO CONECTABLES
    2. SEGURIDAD.

La mayoría de los clientes temen compartir sus datos bancarios y personales en internet, porque esta información puede ser utilizada de mala manera.

* 1. FACILIDAD DE USO
     1. FACTORES HUMANOS

La tienda TecShop será muy fácil de usar para el público, ya que su utilidad se basa principalmente para mejorar las siguientes acciones:

* Los clientes encontraran con facilidad lo que buscan.
* Podrá seleccionar los artículos que desea.
* Podrá consultar precios, detalles y otras opciones de los artículos.
* Mostrará en una lista los artículos que lleva seleccionados.
  1. FIABILIDAD
     1. CAPACIDAD DE RECUPERACIÓN

La capacidad de recuperación hace referencia a la capacidad de restaurar la implementación hasta el momento en que se produjo el error.

* CREACIÓN DE COPIAS DE SEGURIDAD DE LAS BASES DE DATOS: las copias de seguridad son vitales para la capacidad de recuperación de la implementación.
* USO DE ESTÁNDARES DE HARDWARE: el uso de estándares establecidos para el hardware mejora la capacidad de mantenimiento y rendimiento de la implementación.
  1. RENDIMIENTO

Calidad del servicio:

* El servicio al cliente es uno de los elementos más importantes dentro de un negocio en línea, por lo  que medir la efectividad
  1. INTERFACES
     1. INTERFACES Y HARDWARE DESTACABLE
* Computadoras con buenas características
* Sistemas de formas de pago
* Servidores funcionando todas las horas, todos los días del año
  + 1. INTERFACES SOFTWARE

Para los sistemas de colaboración externo como el de facturación, se necesitan de interfaces diferentes.

* 1. REGLAS DE DOMINIO

|  |  |  |  |
| --- | --- | --- | --- |
| ID | REGLA | GRADO DE VARIACION | FUENTE |
| REGLA #1 | Se requiere la manera de pagos mediante tarjetas o depósitos. | La cantidad que sea la requerida para ser válida la compra debe de ser cubierta para poder proceder con el proceso de venta. |  |
| REGLA #2 | El articulo o los artículos tendrán 60 días para cualquier aclaración o devoluciones. | Los productos no se podrán devolver después de pasados los días antes mencionados. |  |

1. **GLOSARIO.**
   1. HISTORIA DE REVISIONES.

|  |  |  |  |
| --- | --- | --- | --- |
| VERSION | FECHA | DESCRIPCION | AUTOR |
| 1ra Versión | 22/10/2016 | Primera versión del glosario, para refinarse durante las próximas fases. | Luis Ramírez |
| 2da Versión | 23/10/2016 | Segunda versión del glosario, con mayor manejo de información. | Mario Sanz |
| 3ra Versión | 27/Octubre/2016 | Refinamiento de la fase de glosario, para tener completa esta fase de inicio. | Mario Sanz |

* 1. DEFINICIONES.

|  |  |  |
| --- | --- | --- |
| TERMINO | DEFINICION | ALIAS |
| ARTICULO | Un artículo en venta en la aplicación. |  |
| AUTORIZACION DE PAGO | Validación llevada a cabo por un servicio externo de autorización de pago que garantizara el pago al vendedor. |  |
| USUARIO | cualquier persona que tenga acceso a la aplicación |  |
| ADMINISTRADOR | persona que tenga acceso autorizado para poder administrar la aplicación |  |
| SOFTWARE | Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas |  |
| HARDWARE | Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático. |  |
| INTERFACE | Dispositivo capaz de transformar las señales generadas por un aparato en señales comprensibles por otro. |  |
| PROCESO UNIFICADO | El Proceso Unificado de Desarrollo Software o simplemente Proceso Unificado es un marco de desarrollo de software que se caracteriza por estar dirigido por | PU |

1. **LISTA DE RIESGOS Y PLAN DE GESTION DE RIESGOS**
   1. LISTA DE RIESGOS

1.- Los problemas financieros de la organización fuerzan a reducir el presupuesto del proyecto

2.- el personal clave está enfermo y no disponible en momentos críticos

3.- los clientes no comprender el impacto de los cambios.

4.- Es ineficiente el código generado por las herramientas CASE

5.- Modificación del cronograma del proyecto

6.- Baja de satisfacción de los interesados del proyecto

7.- metodología inadecuada en el desarrollo del proyecto

* 1. PLAN DE GESTION DE RIESGOS
     1. METODOLOGIA DE GESTION DE RIEGOS

|  |  |  |  |
| --- | --- | --- | --- |
| PROCESO | DESCRIPCION | HERRAMIENTAS | FUENTES DE INFORMACION |
| PLANIFICACION DE GESTION DE RIEGOS | .- Elaborar plan de gestión de riesgos | .- PMBOK | .- sponsor y usuarios  .- PM y equipo de proyecto |
| IDENTIFICACION DE RIESGOS | .- Identificar que riesgos pueden afectar el proyecto y documentar sus características | .- checklist de riesgos  .- listado de riesgos de la compañía | .- sponsor y usuarios  .- PM y equipo de proyecto  .- archivos históricos de proyectos |
| ANALISIS CUALITATIVO DE RIESGOS | .- Evaluar probabilidad e impacto  .- establecer un raking de importancia | .- definición de probabilidad e impacto  .- matriz de probabilidad e impacto | .- sponsor y usuarios  .- PM y equipo de proyecto |
| PLANIFICACION DE LA RESPUESTA DE RIESGOS | .- Definir respuesta a riesgos  .- Planificar ejecución de respuestas |  | .- Sponsor y usuarios  .- PM y equipo de proyecto  .- Archivos históricos de proyectos |
| SEGUIMIENTO Y CONTROL DE RIESGOS | .- Verificar la ocurrencia de riesgos  .- supervisar y verificar la ejecución de respuestas  .- verificar aparición de nuevos riegos |  | .- Sponsor y usuarios  .- PM y equipo de proyecto |

* + 1. ROLES Y RESPONSABILIDADES DE LA GESTIÓN DE RIESGOS

|  |  |  |
| --- | --- | --- |
| PROCESO | ROLES | RESPONSABILIDADES |
| PLANIFICACION DE GESTION DE RIEGOS | EQUIPO DE GESTION DE RIESGOS:  .- Líder, Miembros | Dirigir actividad, responsable directo.  Ejecutar actividad |
| IDENTIFICACION DE RIESGOS | EQUIPO DE GESTION DE RIESGOS:  .- Líder, Miembros | Dirigir actividad, responsable directo.  Ejecutar actividad |
| ANALISIS CUALITATIVO DE RIESGOS | EQUIPO DE GESTION DE RIESGOS:  .- Líder, Miembros | Dirigir actividad, responsable directo.  Ejecutar actividad |
| PLANIFICACION DE LA RESPUESTA DE RIESGOS | EQUIPO DE GESTION DE RIESGOS:  .- Líder, Miembros | Dirigir actividad, responsable directo.  Ejecutar actividad |
| SEGUIMIENTO Y CONTROL DE RIESGOS | EQUIPO DE GESTION DE RIESGOS:  .- Líder, Miembros | Dirigir actividad, responsable directo.  Ejecutar actividad |

* + 1. CALENDARIO DE LA GESTION DE RIESGOS

|  |  |  |  |
| --- | --- | --- | --- |
| PROCESO | MOMENTO DE EJECUCION | PERIOCIDAD | FECHA DE EJECUCION |
| PLANIFICACION DE GESTION DE RIEGOS | .- Al inicio del proyecto | .- Pan de proyecto | .- Una vez |
| IDENTIFICACION DE RIESGOS | .- Al inicio del proyecto  .- En cada reunión del equipo del proyecto | .- Pan de proyecto  .- Reunión de coordinación semanal | .- Una vez por semana |
| ANALISIS CUALITATIVO DE RIESGOS | .- Al inicio del proyecto  .- En cada reunión del equipo del proyecto | .- Pan de proyecto  .- Reunión de coordinación semanal | .- Una vez por semana |
| PLANIFICACION DE LA RESPUESTA DE RIESGOS | .- Al inicio del proyecto  .- En cada reunión del equipo del proyecto | .- Pan de proyecto  .- Reunión de coordinación semanal | .- Una vez por semana |
| SEGUIMIENTO Y CONTROL DE RIESGOS | .- En cada fase del proyecto | .- Reunión de coordinación semanal | .- Una vez por semana |

* + 1. MATRIZ DE PROBABILIDAD E IMPACTO

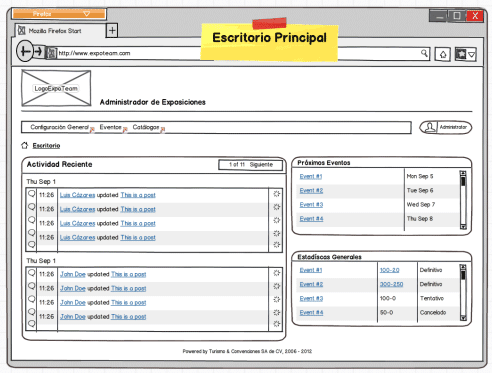
|  |  |  |  |
| --- | --- | --- | --- |
| PROBABILIDAD | VALOR NUMERICO | IMPACTO | VALOR NUMERICO |
| Muy importante | 0.3 | Muy bajo | 0.05 |
| Relativamente probable | 0.3 | bajo | 0.10 |
| Probable | 0.5 | moderado | 020 |
| Muy probable | 0.7 | alto | 0.40 |
| Casi certeza | 0.9 | Muy alto | 0.80 |

|  |  |
| --- | --- |
| TIPO DE RIESGO | PROBABILIDAD x IMPACTO |
| muy alto | Mayor a 0.50 |
| alto | Menor a 0.50 |
| Moderado | Menor a 0.30 |
| Bajo | Menor a 0.10 |
| Muy bajo | Menor a 0.05 |

* + 1. TABLA DE ANALISIS DE RIESGOS

|  |  |  |
| --- | --- | --- |
| RIESGO | PROBABILIDAD | EFECTOS |
| 1.- Los problemas financieros de la organización fuerzan a reducir el presupuesto del proyecto | Baja | Catastrófico |
| 2.- el personal clave esta enfermo y no disponible en momentos críticos | Media | Serio |
| 3.- los clientes no comprender el impacto de los cambios. | Media | Tolerable |
| 4.- Es ineficiente el código generado por las herramientas CASE | Media | Insignificante |
| 5.- Modificación del cronograma del proyecto | Alta | Serio |
| 6.- Baja de satisfacción de los interesados del proyecto | Media | Tolerable |
| 7.- metodología inadecuada en el desarrollo del proyecto | Alta | Catastrófico |

1. **PROTOTIPOS Y PRUEBAS DE CONCEPTOS**
   1. PROTOTIPOS



****

* 1. PRUEBAS DE CONCEPTOS
* Descripción del concepto:

TecShop, Una tienda virtual adecuada a sus necesidades.

* Idea del proyecto:

Satisfacer las necesidades de los clientes con los artículos.

* Información para el comprador:

TecShop va a consistir en una interfaz web dinámica que permita por una parte a los clientes que estén registrados o que la visiten consultar los artículos que estén disponibles, así como todos los datos sobre los artículos, para de esta manera los usuarios puedan mantenerse informados de los artículos en la tienda.

* Idea formalizada:

Catálogo de artículos organizados

Carrito de compra

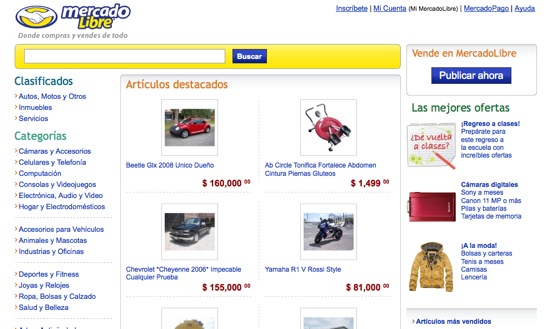
Procesos de forma tradicional

Diferentes maneras de realizar pago

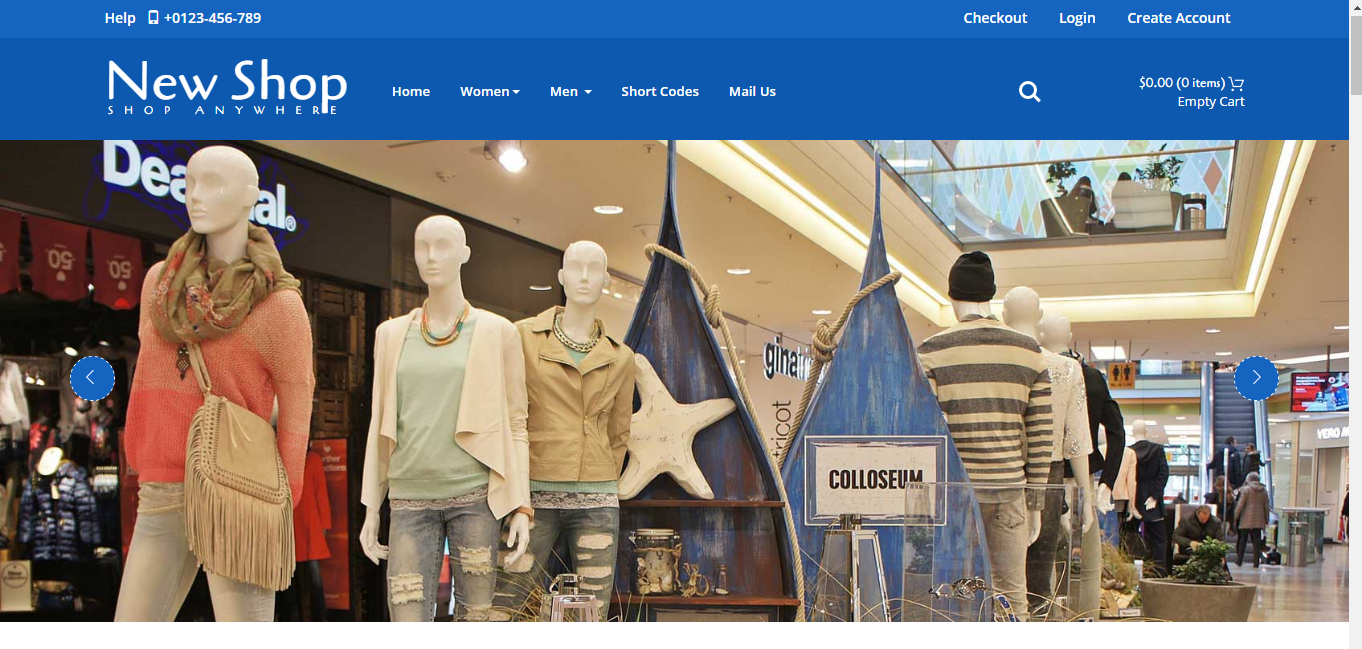
* Descripción atractiva:

Satisfacer las necesidades de los clientes con los artículos, para asegurar una relación con nuestros clientes, obteniendo de esta manera una adecuada rentabilidad y garantizando así nuestra permanencia y el crecimiento de nuestra empresa.

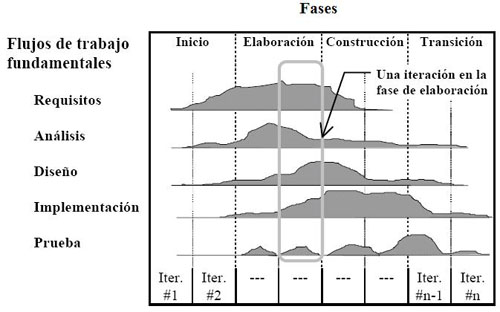
* Representación visual:



* Prototipo de producto final:

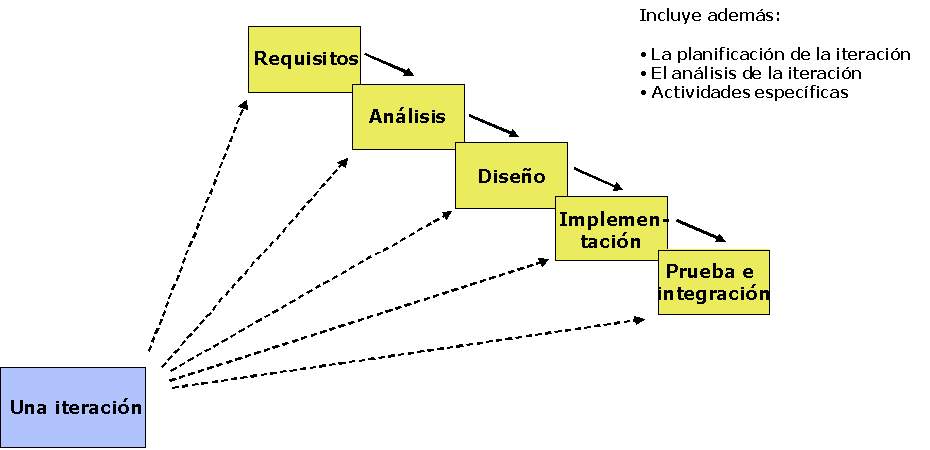


1. **PLAN DE ITERACION**



PROCESO ITERATIVO E INCREMENTAL

* El ciclo de vida iterativo se basa en la evolución de prototipos ejecutables que se muestran a los usuarios y clientes.
* En el ciclo de vida iterativo a cada iteración se reproduce el ciclo de vida en cascada a menor escala.
* Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.
* Las actividades se encadenan en una mini-cascada con un alcance limitado por los objetivos de la iteración



CADA ITERACIÓN COMPRENDE

* Planificar la iteración (estudio de riesgos)
* Análisis de los Casos de Uso y escenarios
* Diseño de opciones arquitectónicas
* Codificación y pruebas. La integración del nuevo código con el existente de iteraciones anteriores se hace gradualmente durante la construcción
* Evaluación de la entrega ejecutable (evaluación del prototipo en función de las pruebas y de los criterios definidos)
* Preparación de la entrega (documentación e instalación del prototipo)
* Cada una de las cuatro fases termina con hito principal.

El número de iteraciones planeado para cada fase depende, básicamente de la complejidad del sistema.

* Fase de Inicio:

Una iteración, principalmente dedicada a definir el ámbito del sistema

* Fase de elaboración:

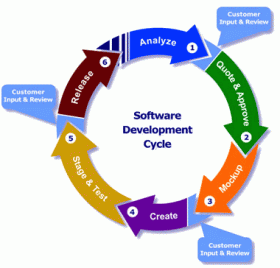
Dos iteraciones, la primera para esbozar la arquitectura y la segunda para completar la línea base de la arquitectura

* Fase de construcción:

Dos iteraciones, para asegurar que los incrementos resultantes funcionan satisfactoriamente

* Fase de transición:

Una iteración



LINEA BASE:

* Conjunto de artefactos revisados y aprobados que constituyen una base convenida para la evolución y desarrollo adicional y que se puede cambiar solamente a través de la administración de cambios.
* Asegurarse qué subsistemas, cuándo alcanzan un nivel especifico de la madurez, son la línea base para que esté disponible para la liberación o la reutilización en iteraciones subsecuentes del proyecto o otros proyectos.
* Se considera como candidato para una Línea Base el conjunto de archivos y directorios bajo control de versión que son desarrollados, integrados y puestos juntos en una liberación.
* Una línea base se crea al final de cada iteración

VERSIONES

* Identifican el estado de un elemento de configuración o una configuración en un punto definido en el tiempo
* Conjunto de artefactos relativamente completo y consistente.

LIBERACIÓN:

* Es una versión que se ha puesto disponible a los usuarios
* Regularmente está asociado a una línea base de una configuración

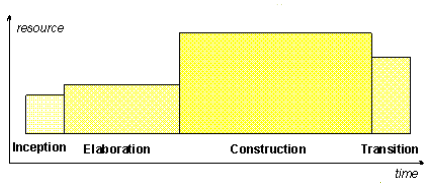
ESFUERZO Y DEDICACION POR FASES EN PU.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | INICIO | ELABORACION | CONSTRUCCION | TRANSICION |
| TIEMPO DEDICADO | 10% | 30% | 50% | 10% |

EJEMPLO DE DURACION (7 Meses).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | INICIO | ELABORACION | CONSTRUCCION | TRANSICION |
| TIEMPO DEDICADO | 10% | 30% | 50% | 10% |
| SEMANAS | 2.8 semanas | 8.4 semanas | 14 semanas | 2.8 semas |

DISTRIBUCION DE RECURSOS POR FASE.



EJEMPLO DE PLAN DE ITERACIONES

|  |  |  |
| --- | --- | --- |
| FASE | N° ITERACIONES | DURACION |
| INICIO | 1 | 3 semanas |
| ELABORACION | 2 | 8 semanas |
| CONSTRUCCION | 3 | 14 semanas |
| TRANSICION | 2 | 3 semanas |

DETALLE DE LAS ITERACIONES

|  |  |  |
| --- | --- | --- |
| FASE | N° ITERACION | DESCRIPCION |
| INICIO | 1 Iteración Preliminar | Definir el modelo de negocio, los productos de requerimientos, el plan de desarrollo de software |
| ELABORACION | 2 Iteraciones para el desarrollo de prototipos de la arquitectura. | Análisis y diseño para todos los casos de uso de las liberaciones. |
| CONSTRUCCION | 3 iteraciones para el desarrollo | Implementar todos los casos de uso para la versión beta. |
| TRANSICION | 2 iteraciones para la implementación. | Instalaciones. |

1. **FASE PLAN Y PLAN DE DESARROLLO DEL SOFTWARE**
   1. PLAN DE DESARROLLO DE SOFTWARE

|  |  |  |
| --- | --- | --- |
| DISCIPLINAS/ARTEFACOS GENERADOS DURANTE LA FASE DE INICIO | COMIENZO | APROBACION |
| MODELADO DEL NEGOCIO |  |  |
| .- Modelado de casos de uso del negocio y modelo de objetos del negocio | Semana 7 | Semana 10 |
| REQUISITOS |  |  |
| .- Visión | Semana 8 | Semana 10 |
| .- Modelo de casos de uso | Semana 9 | Siguiente fase |
| .- Especificación de casos de uso | Semana 9 | Siguiente fase |
| ANALISIS/DISEÑO |  |  |
| .- Modelo análisis/diseño | Semana 8 | Siguiente fase |
| ­.-Modelo de datos | Semana 8 | Siguiente fase |
| IMPLEMENTACION |  |  |
| .- Prototipos de interfaces de usuario | Semana 9 | Siguiente fase |
| PRUEBAS |  |  |
| .- Casos de prueba funcionales | Semana 9 | Siguiente fase |
| DESPLIEGUE |  |  |
| .-Modelo de despliegue | Semana 9 | Siguiente fase |
| GESTION DE CAMBIOS Y CONFIGURACION | Durante todo el proyecto |  |
| GESTION DEL PROYECTO | Semana 7 | Semana 10 |

|  |  |  |
| --- | --- | --- |
| DISCIPLINAS/ARTEFACOS GENERADOS DURANTE LA FASE DE ELABORACION | COMIENZO | APROBACION |
| MODELADO DEL NEGOCIO |  |  |
| .- Modelado de casos de uso del negocio y modelo de objetos del negocio | Semana 7 | Aprobado |
| REQUISITOS |  |  |
| .- Visión | Semana 8 | Aprobado |
| .- Modelo de casos de uso | Semana 9 | Semana 12 |
| .- Especificación de casos de uso | Semana 9 | Semana 12 |
| ANALISIS/DISEÑO |  |  |
| .- Modelo análisis/diseño | Semana 8 | Revisar en cada iteración |
| ­.-Modelo de datos | Semana 8 | Revisar en cada iteración |
| IMPLEMENTACION |  |  |
| .- Prototipos de interfaces de usuario | Semana 9 | Revisar en cada iteración |
| PRUEBAS |  |  |
| .- Casos de prueba funcionales | Semana 9 | Revisar en cada iteración |
| DESPLIEGUE |  |  |
| .-Modelo de despliegue | Semana 9 | Revisar en cada iteración |
| GESTION DE CAMBIOS Y CONFIGURACION | Durante todo el proyecto |  |
| GESTION DEL PROYECTO | Semana 7 | Revisar en cada iteración |

|  |  |  |
| --- | --- | --- |
| DISCIPLINAS/ARTEFACOS GENERADOS DURANTE LA FASE DE CONSTRUCCION  (ITERACION #1) | COMIENZO | APROBACION |
| CASOS DE USO NEGOCIADOS PARA LA PRIMERA LIBERACION |  |  |
| .- Caso de uso escogido | Semana 13 | Semana 14 |

|  |  |  |
| --- | --- | --- |
| DISCIPLINAS/ARTEFACOS GENERADOS DURANTE LA FASE DE CONSTRUCCION  (ITERACION #2) | COMIENZO | APROBACION |
| CASOS DE USO NEGOCIADOS PARA LA PRIMERA LIBERACION |  |  |
| .- Caso de uso escogido | Semana 15 | Semana 16 |

1. **MARCO DE DESARROLLO**

Para el desarrollo del proyecto se seguirá la metodología Proceso Unificado (PU), es un ciclo de vida iterativo e incremental.

PU divide el proceso de desarrollo en los siguientes ciclos:

* Planificación, descripción del proyecto, objetivos plan de trabajo y funcionalidades.
* Análisis de requisitos, identificación de subsistemas, diagramas de casos de uso, prototipos de la aplicación.
* Diseño, representación gráfica de los subsistemas, diagramas de clases y jerarquías en UML, relación de clases, diagramas de colaboración, diseño de la persistencia, variaciones del prototipo de la interface gráfica.
* Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FASE | TAREA | DURACION | COMIENZO | FIN |
| INICIO |  |  |  |  |
|  | DESCRIPCION DEL NEGOCIO | 6 días | Jueves 20/10/16 | Miércoles 26/10/16 |
|  | MODELO DE CASOS DE USO DEL NEGOCIO | 6 días | Martes 25/10/16 | Lunes 31/10/16 |
|  | MODELO DE ANALISIS DEL NEGOCIO | 9 días | Miércoles 01/11/16 | Jueves 10/11/2016 |
| ELABORACION |  |  |  |  |
|  | CAPTURA DE REQUERIMIENTOS | 11 días | Viernes 11/11/2016 | Martes 22/11/2016 |
|  | ANALISIS Y DISEÑO | 21 días | Miércoles 23/11/2016 | Miércoles 14/12/2016 |
| CONSTRUCCION |  |  |  |  |
|  | IMPLEMENTACION | 21 días | Jueves 15/12/2016 | Domingo 15/01/2017 |
|  | PRUEBAS | 11 días | Lunes 16/01/2017 | Viernes 27/01/2017 |
| TRANSICION |  |  |  |  |
|  | PRUEBAS DE ACEPTACION | 11 días | Sábado 28/01/2017 | Jueves 09/02/2017 |
|  | PUESTO EN PRODUCCION |  | Lunes 27/02/2017 |  |