**<Sistema de Control de Entregas>**

**Plan de Desarrollo de Software**

**Version <1.4>**

**Historial**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| <22/septiembre/15> | <1.0> | Introducción, propósito | Michel Martinez  Alejandro Lopez  Jesus Bernardo Borboa Lopez |
| <28/septiembre/15> | <1.1> | objetivo de iteración  calendario  plan de fases | Michel Martinez  Alejandro Lopez  Jesus Bernardo Borboa Lopez |
| 06/diciembre/15 | <1.2> | Corrección de errores | Michel E. Martínez  Jesus Bernardo Borboa López  Alejandro López |
| 28/03/2016 | <1.3> | Actualización del plan, integración de un nuevo release | Borboa Lopez Jesus Bernardo |
| 13/Mayo/16 | <1.4> | Actualización del plan en la parte de las iteraciones. | Michel Martínez  Bernardo Borboa  Alejandro López |

**Tabla de Contenido**

1. Introducción

1.1 Propósito

1.2 Definiciones, Acrónimos, y Abreviaciones

1.3 Referencias

1.4 Resumen

2. Vista del Proyecto

2.1 Propósito, Alcance y Objetivos

2.2 Suposiciones y restricciones

2.3 Entregables del proyecto

2.4 Evolución del plan de desarrollo de software

3. Organización del Proyecto

3.1 Estructura organizacional

3.2 Interfaces Externas

3.3 Roles y Responsabilidades

4. Administración del Proceso

4.1 Plan del Proyecto

4.1.1 Plan de Fases

4.1.2 Objetivos Iteración

4.1.3 Releases

4.1.4 Calendario del Proyecto

4.1.5 Recursos del Proyecto

4.1.6 Presupuesto

4.2 Plan de Iteraciones

4.3 Seguimiento y control del Proyecto

4.3.1 Plan de Administración de Requerimientos

4.3.2 Plan de control de plazos

4.3.3 Budget Control Plan

4.3.4 Plan de Control de Calidad

4.4 Plan de Manejo de Riesgos

5. Technical Process Plans

5.1 Development Case

5.2 Methods, Tools, and Techniques

5.3 Infrastructure Plan

5.4 Product Acceptance Plan

6. Plan de Procesos de Soporte

6.1 Plan de Administración de la Configuración

6.2 Evaluation Plan

6.3 Documentation Plan

6.4 Quality Assurance Plan

6.5 Problem Resolution Plan

6.6 Subcontractor Management Plan

6.7 Process Improvement Plan

7. Additional Plans

8. Annexes

9. Index

**Software Development Plan**

1. **Introducción**

Este documento provee una visión global del enfoque de desarrollo propuesto, se centra principalmente en el desarrollo del software, los procesos realizados por otras áreas de la organización van en documentos por separado, aquí solo se mencionarán.

En este documento se describen roles y actividades a realizar por cada uno, entregables por cada fase del desarrollo, y tiempos para cada actividad, además de información del equipo de trabajo y la descripción del producto final que se entregará.

* 1. **Propósito**

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto, tomando en cuenta todos los aspectos importantes.

El Jefe del Proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento. Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ellos**.**

* 1. **Definiciones, Acrónimos, y Abreviaciones**

Verificar en el documento Glosario.

* 1. **Referencias**
  2. **Resumen**

Descripción del Producto — proporciona una descripción del propósito, alcance y objetivos del proyecto.

Entregables --- establece los artefactos que serán producidos y utilizados durante el proyecto.

Equipo de Trabajo  — describe la estructura organizacional del equipo de desarrollo.

Gestión del Proyecto  — explica la planificación estimada, define las fases de RUP y MoProsoft, incluye el calendario resumido, el detalle se manejará a través de MSProject.

1. **Vista del Proyecto**
   1. **Propósito, Alcance y Objetivos**

El propósito del proyecto del Sistema de control de entregas es llevar a cabo un control del proceso de entregas de la empresa Landsberg en el cual los choferes deberán generar una bitácora diaria referente a los pedidos que entregó en una jornada laboral y con la información obtenida de las bitácoras diarias generar estadísticas de tiempo promedio de rutas, kilometraje recorrido y promedio de ordenes.

El alcance es que con el Sistema de control de entregas el supervisor sea capas de tomar decisiones sobre que rutas son mas eficientes además de comprometer a los choferes a realizar su trabajo eficientemente y así tener mayor productividad en la jornada laboral.

El objetivo del Sistema de control de entregas es minimizar el gasto en recursos como es el combustible y realizar mayor número de entregas de mercancía en el menor tiempo posible.

* 1. **Suposiciones y restricciones**
* Restricciones de diseño e instrucción:

Definidas por el cliente.

* Legales y reglamentarios:

Necesidades impuestas por leyes y reglamentos.

* 1. **Entregables del proyecto**

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables, basados en RUP y MoProSoft.

Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos

1. **Plan de Desarrollo del Software.**

Es el presente documento.

1. **Modelado de Negocio.**

Es un modelo de las funciones de negocio, permite situar al sistema en el contexto organizacional.

1. **Visión.**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Proporciona una base contractual más detallada en cuanto a los requisitos técnicos del sistema.

1. **Especificaciones de Requerimientos.**

Describe de manera detallada la funcionalidad del sistema representada inicialmente a través del modelo de casos de uso. Cada caso de uso debe ser especificado mediante un documento que incluye flujos de eventos básicos y alternos, pre condiciones, post condiciones, y requisitos especiales (requisitos no-funcionales asociados). Las funcionalidades deben ser adecuadas, exactas y seguras.

1. **Glosario.**

Es un documento que define los principales términos usados en el proyecto, proporcionando una terminología consistente que ayuda a evitar los malos entendidos.

1. **Prototipos de Interfaces de Usuario**

Definición de aquellas características de la interfaz de usuario que permiten que el software sea fácil de entender, aprender, que genere satisfacción y con el cual el usuario pueda desempeñar su tarea eficientemente, incluyendo la descripción del prototipo de la interfaz. El propósito principal de crear un prototipo del usuario-interfaz es poder exponer y probar la funcionalidad y la utilidad del sistema antes del comienzo verdadero del diseño y del desarrollo.

1. **Análisis y Diseño: Arquitectónica y Detallada**

Este documento tiene la descripción textual y gráfica de la estructura de los componentes de software. El cual consta de las siguientes partes:

Arquitectónica:

Contiene la estructura interna del sistema, es decir la descomposición del sistema en subsistemas. Así como la identificación de los componentes que integran los subsistemas y las relaciones de integración entre ellos. Se compone por Diagramas de Clases, Objetos, Despliegue; el pseudocódigo con la descripción detallada de clases y métodos; sólo por mencionar algunos componentes.

Detallada:

Contiene el detalle de los componentes que permita de manera evidente su construcción y prueba en el ambiente de programación.

1. **Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por un a base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para ello, se utilizará un diagrama entidad-relación que represente a las tablas, campos, campos llave y relaciones entre éstas.

1. **Plan de Pruebas de Integración.**

Descripción que contiene:

El orden de integración de los componentes o subsistemas, guiado por la parte arquitectónica del *Diseño.*

Pruebas que se aplicarán para verificar la interacción entre los componentes.

1. **Plan de Pruebas de Sistema.**

Identificación de pruebas requeridas para el cumplimiento de los requerimientos especificados.

1. **Componentes.**

Conjunto de unidades de código relacionadas.

1. **Software.**

Sistema de software, destinado a un cliente o usuario, constituido por componentes agrupados en subsistemas, posiblemente anidados.

1. **Modelo de Implementación**

Este modelo es una colección de componentes y los subsistemas que los contienen. Los componentes incluyen entregables, tales como ejecutables, y los componentes de los cuales se producen los entregables, por ejemplo archivos del código de fuente.

1. **Modelo de Despliegue**

Este modelo muestra el despliegue la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

1. **Reporte de Pruebas del Sistema.**

Registro de participantes, fecha, lugar, duración y de defectos encontrados.

1. **Reporte de Pruebas de Integración.**

Registro de participantes, fecha, lugar, duración y de defectos encontrados.

1. **Manual de Usuario.**

Documento electrónico o impreso que describe la forma de uso del software con base a la interfaz del usuario. Éste deberá ser redactado en términos comprensibles a los usuarios.

1. **Manual de Operación.**

Documento electrónico o impreso que contenga la información indispensable para la instalación y administración del software, así como el ambiente de operación (sistema operativo, base de datos, servidores, etc.). Éste deberá ser redactado en términos comprensibles al personal responsable de la operación.

1. **Manual de Mantenimiento.**

Documento electrónico o impreso que describe la *Configuración de Software* y el ambiente usado para el desarrollo y pruebas (compiladores, herramientas de análisis y diseño, construcción y pruebas). Este deberá ser redactado en términos comprensibles al personal de mantenimiento.

1. **Registro de Rastreo.**

Relación entre los requerimientos, elementos análisis y diseño, componentes y planes de pruebas.

1. **Solicitud de cambio.**

Las solicitudes de cambio se utilizarán para documentar las peticiones formales propuestas por los usuarios ya sea por ampliaciones o defectos. Su propósito es rastrear todos los requerimientos relacionados a lo largo del ciclo de vida del proyecto.

1. **Configuración del Software**

Conjunto consistente de productos de software que incluye:

Especificación de Requerimientos

Análisis y Diseño

Software

Registro de Rastreo

Plan de pruebas del sistema

Reporte de pruebas del sistema

Plan de integración de pruebas

Reporte de pruebas de integración

Manual de usuario

Manual de operación

Manual de mantenimiento

* 1. **Evolución del plan de desarrollo de software**

El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.

1. **Organización del Proyecto** 
   1. **Estructura organizacional**

Administración de Proyectos(RAPE)

Analista

Diseñador

Programador

* 1. **Interfaces Externas**

No aplica.

* 1. **Roles y Responsabilidades**

|  |  |
| --- | --- |
| **Rol** | **Responsabilidad** |
| *RAPE* | *-Ser líder en la toma de decisiones , planificación del proyecto, supervisión del equipo de trabajo.* |
| Analista | Tener el conocimiento y experiencia en la obtención, especificación y análisis de los requerimientos del cliente. |
| Diseñador | Tener el conocimiento y experiencia en el diseño de la estructura de los componentes de software. |
| Programador | Tener el conocimiento y/o experiencia en la programación, integración y pruebas unitarias. |

1. **Administración del Proceso**
   1. **Plan del Proyecto**
      1. *Plan de Fases*

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase

|  |  |  |
| --- | --- | --- |
| **Fase** | **Número**  **Iteraciones** | **Duración** |
| Fase de Inicio | 1 | 1 semana |
| Fase de Elaboración | 3 | 6 semanas |
| Fase de Construcción | 3 | 18 semanas |
| Fase de Transición | 3 | 7 semana |

* + 1. *Objetivos Iteración*

|  |  |  |  |
| --- | --- | --- | --- |
| **Descripción** | **Hito** | **Objetivos** | **Riesgos asociados** |
| Fase de Inicio | En esta fase se generará la documentación necesaria para describir las actividades de los procesos de entregas de la empresa Landsberg que se detecten al momento de realizar el análisis del negocio.  Además, desarrollarán las características del producto desde la perspectiva del usuario, que apoye la realización de las actividades los cuales serán establecidas en los artefactos Visión y en la especificación de requerimientos.  . | En esta etapa se realizan las actividades de MoProSoft A1 Realización de la fase de Inicio y A2 Realización de la fase de Requerimientos.  Se revisará el Plan de Desarrollo de Software (este documento) para lograr un entendimiento de lo que se va a realizar, entre los miembros del equipo de trabajo y entre el equipo de trabajo y el cliente. En esta etapa se realiza el modelado del negocio donde quedan establecidos los procesos que se desarrollan en la empresa. Se establecen los requerimientos y las bases de la arquitectura que se desarrollará más adelante, deben quedar claras todas las actividades que deberá realizar el sistema.  Como entregables en esta fase podemos mencionar:   * Plan de desarrollo de Software. * Documento de visión. * Especificación de Requerimientos. * Modelado de Negocio. * Glosario. * Reporte de Actividades de esta fase * Reportes de verificación y validación. * Lecciones aprendidas. * Solicitud de cambio. | Crear un plan de desarrollo de acuerdo al tiempo real del proyecto.  Comprender los requerimientos del usuario.  Definir la solución a la problemática presentada por el cliente.  Determinar la factibilidad del proyecto desde el punto de vista del negocio.  Establecer la comunicación con el cliente. |
| Fase de Elaboración | En esta fase los principales casos de uso serán identificados, y en base a esto se generarán documentos que indiquen la funcionalidad de cada caso de uso. Se analizan los requisitos y se desarrolla un prototipo de arquitectura. Posteriormente se realizará prototipos de interfaz los cuales se realizarán de tal manera que sean utilizables en la fase de construcción.  La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase.  La primera iteración tendrá como objetivo la especificación de los casos de uso de la release 1,así como su realización preliminar en el Modelo de Análisis/Diseño, Y la actualización de los requerimientos de ser necesaria, también se realizarán los prototipos de interfaz preliminares relacionados con el release 1.  La segunda iteración tendrá el objetivo de identificar todos los casos de uso especificados en el release 2, así como el de crear los prototipos de interfaz preliminares relacionados con el release 2. De igual forma permitirá hacer una revisión general del estado de los artefactos hasta este punto, y actualizar si es necesario los artefactos con el fin del cumplimiento de los objetivos planteados.  La tercera iteración consistirá en buscar y corregir problemas en el diseño y construcción.  Las dos primeras iteraciones tardaran 3 semanas y la tercera iteración durará 1 semana.  La cuarta iteración consistirá en encontrar defectos y errores en los documentos del proyecto y solucionarlos. | En esta etapa se realizará la actividad de MoProSoft A3 Realización de la fase de Análisis y Diseño.  En esta fase de elaboración se utilizará para el diseño, en esta etapa se analizan los requerimientos establecidos en la fase de inicio para producir una descripción de la estructura de los componentes de software, la cual servirá de base para la construcción, se realiza el diseño de la arquitectura y el prototipo de la interfaz.  Como entregables en esta fase podemos mencionar:   * Diseño de Interfaz (Prototipo). * Diseño de Arquitectura (clases de entidad, límite y control, objetos, métodos, subsistemas, etc.) * Documento de Arquitectura de Software. * Manual de usuario preliminar. * Plan de Pruebas de Integración * Registro de Rastreo. * Configuración de software. * Reporte de Actividades de esta fase * Reportes de verificación y validación. * Lecciones aprendidas. * Solicitud de cambio.   Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera y segunda release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis/Diseño). | Clarificar los aspectos de la arquitectura  Riesgos técnicos del hardware |
| Fase de Construcción | La fase de construcción responde con énfasis al desarrollo propiamente de los componentes del sistema, cubriendo el desarrollo del Release 1 y 2, correspondientes a las  iteraciones E1 y E2.  Posteriormente se aplicarán pruebas al sistema, de tal manera que sea evaluado para tomar como salida el primer release.  La primera iteración tendrá como objetivo la refinación de las clases que afectarán al sistema obtenidas en E1, el modelo de entidad-relación y de las interfaces de usuario ya elaboradas. Se iniciará con la construcción de los componentes de los casos de uso principales, los cuales se manejan en el release 1.  La segunda iteración tendrá como objetivo la construcción de los componentes del release 2, así como de la elaboración de los manuales de usuario y manual de operación.  En la tercera iteración se corregirán errores en el código y se buscara mejorar las interfaces de los usuarios  En la cuarta iteración terminar los pendientes de la construcción y arreglos estéticos. | En esta etapa se realizan las actividades de MoProSoft A4 Realización de la Fase de Construcción y A5 Realización de la Fase de Integración y Pruebas.  En esta fase se construyen e integran los componentes de software resultantes del análisis y diseño y se realizan las pruebas unitarias. Su resultado son los componentes de software probados.  Como entregables en esta fase podemos mencionar:   * Manual de usuario final * Guía de Instalación y Configuración * Componentes. * Software. * Reporte de Pruebas de sistema. * Reporte de Pruebas de Integración. * Registro de Rastreo * Configuración de software. * Reporte de Actividades de esta fase. * Reportes de verificación y validación. * Lecciones aprendidas. * Solicitud de cambio. | Riesgos en la interpretación de documentos asociados a la programación  Contar con las herramientas necesarias para el desarrollo del sistema  Generar un manual de usuario entendible por el cliente |
| Fase de Transición | En esta fase durante la primera iteración T1 se empaquetará e instalará la versión delta del sistema que contendrá el R1.0. Se capacitará a los usuarios habiendo elaborado un manual y plan de capacitación y se esperará un tiempo de prueba de 2 semanas.  En la segunda iteración T2 se empaquetará e instalará la versión beta del sistema. Se tendrá un tiempo de prueba de 1 semana. Una vez aceptado se llevará a cabo también su respectiva capacitación y actualización de los manuales correspondientes.  En la tercera transición se buscar probar las mejoras echas y buscar errores en el programa  En la cuarta iteración cargar el sistema en un servidor web que soporte el lenguaje de programación. |  | Contar con el equipo de hardware y software necesario para la transición del sistema hacia el cliente.  Problemas con la migración del servidor local a un servidor de negocio. |

* + 1. *Releases*

El plan de releases está sujeto a cambios de acuerdo al progreso del proyecto. Este puede cambiar de acuerdo a factores de negocio o técnicos. Se utilizará RequisitePro para administrar los requerimientos del producto y mantener la trazabilidad de los contenidos de cada release. En particular, la prioridad y dificultad de los atributos son usados para determinar la prioridad de los requerimientos del producto y su ubicación en cada release.

Plan de Releases con sus respectivas funcionalidades básicas:

R1:

* + Login
  + Mantenimiento a usuarios
  + Mantenimiento a Bitácora

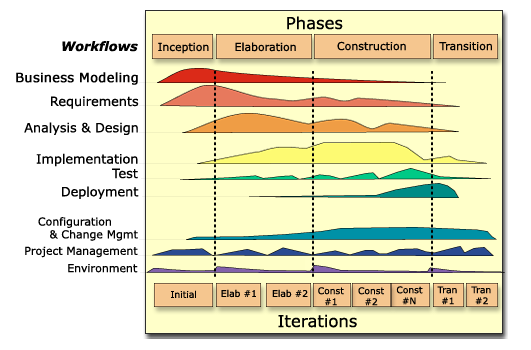
R2:

* + Consultar reporte

R3

* + Software corregido y mejorado
    1. *Calendario del Proyecto*

A continuación se presenta un calendario de las principales tareas del proyecto. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización simultánea de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada flujo de trabajo (workflow) en un momento determinado del desarrollo.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Disciplinas/Artefactos generados o modificados**  **durante la Fase de Inicio- Iteración I1** | **Duración** | **Inicia** | **Termina** | **Rol Responsable** | **Estado** |
| **Modelado del Negocio** |  |  |  |  |  |
| Modelo de Actividades |  |  |  |  |  |
| **Requisitos** |  |  |  |  |  |
| Glosario |  |  |  |  |  |
| Visión |  |  |  |  |  |
| Modelo caso uso |  |  |  |  |  |
| Especificación de Requerimientos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Especificaciones Suplementarias |  |  |  |  |  |
| **Análisis/Diseño** |  |  |  |  |  |
| Modelo de Análisis/Diseño |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Modelo de Datos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Prototipos de Interfaces de Usuario |  |  |  |  |  |
| **Implementación** |  |  |  |  |  |
| Modelo de Implementación |  |  |  |  |  |
| Diagrama de subsistemas (paquetes) |  |  |  |  |  |
| Componente 1 |  |  |  |  |  |
| Componente 2 |  |  |  |  |  |
| Software |  |  |  |  |  |
| **Pruebas** |  |  |  |  |  |
| Plan de pruebas |  |  |  |  |  |
| CPU1 |  |  |  |  |  |
| CPU2 |  |  |  |  |  |
| Plan de pruebas del sistema |  |  |  |  |  |
| **Despliegue** |  |  |  |  |  |
| Modelo de Despliegue |  |  |  |  |  |
| **Gestión de Cambios y Configuración** |  |  |  |  |  |
| **Gestión del Proyecto** |  |  |  |  |  |
| Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Disciplinas/Artefactos**  **generados o modificados durante la**  **Fase de Elaboración– Iteración E1 (4 semanas de duración)** | **Duración** | **Inicia** | **Termina** | **Rol Responsable** | **Estado** |
| **Modelado del Negocio** |  |  |  |  |  |
| Modelo de Actividades |  |  |  |  |  |
| **Requisitos** |  |  |  |  |  |
| Glosario |  |  |  |  |  |
| Visión |  |  |  |  |  |
| Modelo caso uso |  |  |  |  |  |
| Especificación de Requerimientos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Especificaciones Suplementarias |  |  |  |  |  |
| **Análisis/Diseño** |  |  |  |  |  |
| Modelo de Análisis/Diseño |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Modelo de Datos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Prototipos de Interfaces de Usuario |  |  |  |  |  |
| **Implementación** |  |  |  |  |  |
| Modelo de Implementación |  |  |  |  |  |
| Diagrama de subsistemas (paquetes) |  |  |  |  |  |
| Componente 1 |  |  |  |  |  |
| Componente 2 |  |  |  |  |  |
| Software |  |  |  |  |  |
| **Pruebas** |  |  |  |  |  |
| Plan de pruebas |  |  |  |  |  |
| CPU1 |  |  |  |  |  |
| CPU2 |  |  |  |  |  |
| Plan de pruebas del sistema |  |  |  |  |  |
| **Despliegue** |  |  |  |  |  |
| Modelo de Despliegue |  |  |  |  |  |
| **Gestión de Cambios y Configuración** |  |  |  |  |  |
| **Gestión del Proyecto** |  |  |  |  |  |
| Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Disciplinas/Artefactos**  **generados o modificados durante la**  **Fase de Contrucción– Iteración C1 (4 semanas de duración)** | **Duración** | **Inicia** | **Termina** | **Rol Responsable-Nombre** | **Estado** |
| **Modelado del Negocio** |  |  |  |  |  |
| Modelo de Actividades |  |  |  |  |  |
| **Requisitos** |  |  |  |  |  |
| Glosario |  |  |  |  |  |
| Visión |  |  |  |  |  |
| Modelo caso uso |  |  |  |  |  |
| Especificación de Requerimientos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Especificaciones Suplementarias |  |  |  |  |  |
| **Análisis/Diseño** |  |  |  |  |  |
| Modelo de Análisis/Diseño |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Modelo de Datos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Prototipos de Interfaces de Usuario |  |  |  |  |  |
| **Implementación** |  |  |  |  |  |
| Modelo de Implementación |  |  |  |  |  |
| Diagrama de subsistemas (paquetes) |  |  |  |  |  |
| Componente 1 |  |  |  |  |  |
| Componente 2 |  |  |  |  |  |
| Software |  |  |  |  |  |
| **Pruebas** |  |  |  |  |  |
| Plan de pruebas |  |  |  |  |  |
| CPU1 |  |  |  |  |  |
| CPU2 |  |  |  |  |  |
| Plan de pruebas del sistema |  |  |  |  |  |
| **Despliegue** |  |  |  |  |  |
| Modelo de Despliegue |  |  |  |  |  |
| **Gestión de Cambios y Configuración** |  |  |  |  |  |
| **Gestión del Proyecto** |  |  |  |  |  |
| Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Disciplinas/Artefactos**  **generados o modificados durante la**  **Fase de Transición – Iteración T1 (4 semanas de duración)** | **Duración** | **Inicia** | **Termina** | **Rol Responsable** | **Estado** |
| **Modelado del Negocio** |  |  |  |  |  |
| Modelo de Actividades |  |  |  |  |  |
| **Requisitos** |  |  |  |  |  |
| Glosario |  |  |  |  |  |
| Visión |  |  |  |  |  |
| Modelo caso uso |  |  |  |  |  |
| Especificación de Requerimientos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Especificaciones Suplementarias |  |  |  |  |  |
| **Análisis/Diseño** |  |  |  |  |  |
| Modelo de Análisis/Diseño |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Modelo de Datos |  |  |  |  |  |
| CU1 |  |  |  |  |  |
| CU2 |  |  |  |  |  |
| Prototipos de Interfaces de Usuario |  |  |  |  |  |
| **Implementación** |  |  |  |  |  |
| Modelo de Implementación |  |  |  |  |  |
| Diagrama de subsistemas (paquetes) |  |  |  |  |  |
| Componente 1 |  |  |  |  |  |
| Componente 2 |  |  |  |  |  |
| Software |  |  |  |  |  |
| **Pruebas** |  |  |  |  |  |
| Plan de pruebas |  |  |  |  |  |
| CPU1 |  |  |  |  |  |
| CPU2 |  |  |  |  |  |
| Plan de pruebas del sistema |  |  |  |  |  |
| **Despliegue** |  |  |  |  |  |
| Modelo de Despliegue |  |  |  |  |  |
| **Gestión de Cambios y Configuración** |  |  |  |  |  |
| **Gestión del Proyecto** |  |  |  |  |  |
| Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones |  |  |  |  |  |

* + 1. *Recursos del Proyecto*
       1. Plan de contratación

El personal del departamento de desarrollo tiene poco personal de desarrollo y diseño para conocer las necesidades del proyecto. Por lo que, se piensa en contratar a personal con experiencia mínima de 2 años en java, Linux y MySQL

* + - 1. Plan de Adquisición de Recursos
      2. Plan de capacitación

Se capacitará al personal para que fortalezcan sus conocimientos y habilidades en:

* + Análisis y Diseño Orientado a Objetos
  + Introducción al RUP
  + Programación avanzada JAVA

* + 1. *Presupuesto*
  1. **Plan de Iteraciones**
  2. **Seguimiento y control del Proyecto**
     1. *Plan de Administración de Requerimientos*

Los requisitos del sistema son especificados en los artefactos de Visión y Casos de Uso. Cada requisito tendrá una serie de atributos tales como prioridad, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán administrados mediante una solicitud de cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de administración de la configuración y cambios.

* + 1. *Plan de control de plazos*

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el Jefe de proyecto, Responsable de Desarrollo y Mantenimiento, personal involucrado de acuerdo al avance del proyecto y en caso de situaciones que lo ameriten también se involucrará al Grupo Directivo.

* + 1. *Plan de Control de Calidad*

Los defectos detectados en las revisiones y formalizados también en una solicitud de cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias.

* 1. **Plan de Manejo de Riesgos**

1. **Technical Process Plans**
   1. **Development Case**

*[***Methods, Tools, and Techniques**

* 1. **Infrastructure Plan**
  2. **Product Acceptance Plan**

1. **Plan de Procesos de Soporte**
   1. **Plan de Administración de la Configuración**

Se realizará una gestión de configuración para llevar un registro de los artefactos generados y sus versiones. También se incluirá la gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto. Al final de cada iteración se establecerá una baseline (un registro del estado de cada artefacto, estableciendo una versión), la cual podrá ser modificada sólo por una Solicitud de Cambio aprobada.

* 1. **Evaluation Plan**
  2. **Documentation Plan**
  3. **Quality Assurance Plan**
  4. **Problem Resolution Plan**
  5. **Subcontractor Management Plan**
  6. **Process Improvement Plan**

1. **Planes Adicionales**
2. **Anexos**
3. **Índices**