SysTree

Plan de Desarrollo de Software

Versión 0.1

Historial de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 13/10/2019 | 0.1 | Versión preliminar propuesta para el plan de desarrollo de software | Angilberto Caro Zavala |

Tabla de Contenidos

1. Introducción 4

1.1 Propósito 4

1.2 Alcance 4

1.3 Resumen 4

2. Vista General del Proyecto 5

2.1 Propósito, Alcance y Objetivos 5

2.2 Suposiciones y Restricciones 5

2.3 Entregables del proyecto 5

2.4 Evolución del Plan de Desarrollo del Software 7

3. Organización del Proyecto 8

3.1 Participantes en el Proyecto 8

3.2 Interfaces Externas 8

3.3 Roles y Responsabilidades 8

4. Gestión del Proceso 10

4.1 Estimaciones del Proyecto 10

4.2 Plan del Proyecto 10

4.2.1 Plan de las Fases 10

4.2.2 Calendario del Proyecto 11

4.3 Seguimiento y Control del Proyecto 14

Plan de Desarrollo del Software

# 

# Introducción

El presente plan de desarrollo está en una versión preliminar redactada con el fin de incluirse en la propuesta formalmente elaborada para el proyecto SysTree, que nos brinda de un sistema de control de áreas verdes, este documento nos prepara una vista global del desarrollo propuesto.

El enfoque de desarrollo que se propone en este documento forma parte de la configuración del proceso RUP (Rational Unified Process). Se describirán los roles de los participantes, se redactarán las actividades que se realizarán y se mencionarán los entregables (artefactos) que se generarán en el desarrollo, destacando que este mismo plan de desarrollo es uno de los artefactos de RUP.

## Propósito

El propósito de este plan de desarrollo es tener la información necesaria para poder tener bajo control el proyecto SysTree. Se describirá a detalle el enfoque de desarrollo del software. Poder comprender el enfoque del desarrollo y tener el conocimiento necesario del proyecto será de gran utilidad para el jefe del proyecto, quien podrá tener una buena organización de tiempo y recursos, así mismo será capaz de dar seguimiento a las actividades del desarrollo. Así mismo el equipo de desarrollo podrá comprender qué debe hacer, cuándo lo debe hacer y qué actividades están bajo su responsabilidad.

## Alcance

El plan de desarrollo explica y describe partes claves para el desarrollo del sistema SysTree, brinda de información relevante para la comprensión del enfoque, las actividades, iteraciones, procesos, requisitos, y demás artefactos requeridos por el proceso RUP.

## Resumen

A partir de este punto el documento se concentrará en explicar detalladamente el proyecto en las siguientes secciones:

Vista General del Proyecto: Descripción del propósito, alcance y objetivos de este proyecto, recordando los artefactos que se generarán para cumplir con el enfoque del proyecto.

Organización del Proyecto: Mostrará la estructura que tiene el proyecto respecto a los miembros del equipo de trabajo y sus respectivos roles.

Gestión del Proceso: Explica la planeación de tiempos y recursos, se definirán las fases necesarias para el desarrollo y cómo se le dará seguimiento a las mismas.

Planes y Guías de aplicación: Nos proporciona una vista global de los métodos, herramientas y técnicas que se emplearán en el desarrollo del proyecto.

# Vista General del Proyecto

## Propósito, Alcance y Objetivos

### Propósito:

Actualmente existen buenas practicas en el cuidado de las áreas verdes, nosotros buscamos brindar una herramienta que mejore el control y cuidado de las plantas, poniendo a disposición de los usuarios tecnologías de geolocalización y diseño de áreas verdes.

### Alcance:

El alcance de este proyecto se limita a la mejora en el control y gestión de las plantas dentro de un área verde, mediante herramientas que faciliten los cuidados y mantenimientos básicos requeridos por el área verde, lo cual dará como resultado una mejor calidad de vida para las plantas y el ecosistema.

Nos enfocaremos principalmente en los siguientes puntos:

* Buena distribución de plantas en el área mediante una herramienta de diseño.
* Asignación de tareas relacionadas con los cuidados de las plantas.
* Buen control de las plantas del área verde mediante herramientas de geolocalización.

### Objetivos:

El objetivo del proyecto es aumentar el numero de plantas y su calidad de vida dentro de un área verde.

## Suposiciones y Restricciones

* Uso de tecnologías de Geolocalización.
* Uso de tecnologías de almacenamiento NO-SQL.
* Uso de tecnologías de Microsoft tales como Bing Maps y .Net Core.
* Uso de la metodología RUP.
* Uso de la herramienta Enterprise Architect para la formulación de los artefactos de RUP.

## Entregables del proyecto

A continuación, se describen los artefactos que se generarán a lo largo del desarrollo del proyecto, la lista que se presenta esta echa conforme a la configuración de entregables requeridas desde el punto de vista de la metodología RUP.

Es importante destacar que de acuerdo con RUP, los artefactos de software generados están sujetos a cambios, de tal forma que solo al final del proyecto se tendrá una versión final y completa de todos los entregables. Sin embargo, En cada iteración e hito del proyecto se desea alcanzar un nivel de completitud necesaria para darle estabilidad tanto al proyecto como a cada uno de los entregables.

1. **Plan de Desarrollo del Software**

Es el presente documento, que se encuentra en su versión preliminar.

1. **Modelo de Casos de Uso del Negocio**

Es un modelo de las funciones que contendrá el sistema desde el punto de vista de los actores externos involucrados en el negocio (Agentes de registro, solicitantes, otros sistemas etc.). Dicho modelo se representa con un diagrama de casos de uso en las que se relaciona a los actores involucrados con las funciones relevantes del sistema.

1. **Modelo de Objetos del Negocio**

Modelo donde se describe cómo se realizarán los casos de uso mostrando cómo interactúan los actores con las funciones del sistema los diagramas de secuencia que generan los casos de uso. Para representar este modelo se utilizan Diagramas de Colaboración Diagrama de Clases, Diagramas de Actividad.

1. **Glosario**

Documento en el que se explican términos importantes en el sistema para la comprensión del cliente, usuario y equipo de desarrollo.

1. **Modelo de Casos de Uso**

Diagrama donde se representan los actores y su relación con las principales funciones del sistema.

1. **Visión**

Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema. Determinando la viabilidad del proyecto.

1. **Especificaciones de Casos de Uso**

Para los casos de uso se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, postcondiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación grafica mediante un Diagrama de actividades.

1. **Especificaciones Adicionales**

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.

1. **Prototipos de Interfaces de Usuario**

Se trata de prototipos que permiten al usuario hacerse una idea mas clara y precisa de las interfaces que tendrá el sistema y así, conseguir feedback respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta grafica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo con el avance del proyecto. Solo los prototipos ejecutables iterativos serán entregados al final de la fase de elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de construcción en la medida que el resultado de las interacciones vaya desarrollando el producto final.

1. **Modelo de Análisis y Diseño**

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de que debe hacerse (sin incluir aspectos de implementación) hacia el cómo se debe hacer (incluyendo una orientación hacia el entorno de implementación), de acuerdo con el avance del proyecto.

1. **Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por una base de datos no relacional, este utiliza un modelo de almacenamiento que está optimizado para los requisitos específicos del tipo de datos que se almacenará, de acuerdo con el modelo no relacional.

1. **Modelo de Implementación**

Este modelo es un conjunto de componentes y subsistemas que forman la unión física de la implementación del sistema. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y otros tipos de ficheros necesarios para la implementación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

1. **Modelo de Despliegue**

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

1. **Casos de Prueba**

Los casos de prueba establecen las condiciones de ejecución, las entradas de las pruebas y los resultados esperados y se especifican mediante un documento. Estos casos de prueba son aplicados como pruebas de regresión en cada interacción. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y dependiendo del tipo de prueba (unitarias, de integración, de componentes, de sistema, etc.) dicho procedimiento podrá ser automatizable mediante un script de prueba.

1. **Solicitud de Cambio**

Los cambios propuestos para los artefactos de formalizan mediante este documento. Por medio de este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. Así se proporciona un registro de decisiones de cambios, de su evaluación e impacto, y se asegura que estos sean conocidos por el equipo de desarrollo. Los cambios se establecen mediante la última baseline (el estado del conjunto de los artefactos en un momento determinado del proyecto) establecida. al final de cada iteración se establecerá una baseline.

1. **Plan de Iteración**

Es un conjunto de actividades ordenadas en base al tiempo que tiene cada actividad, con recursos asignados, dependencias entre ellas. Se realiza para cada iteración, y para todas las fases. Sabiendo que una iteración debe identificar las actividades, los riesgos involucrados, los artefactos a actualizar o crear, las necesidades de adquisición de bienes o servicios.

1. **Evaluación de Iteración**

Este documento incluye le evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados. Considerando que las actividades sean realizadas de manera correcta y supervisadas para que cumplan con los requisitos y no sean desviados del plan de desarrollo de software.

1. **Lista de Riesgos**

En este documento se incluye una lista de las probabilidades de impacto que se conocen y que están vigentes en el proyecto, ordenados en orden descendiente de importancia y con acciones específicas en caso de que ocurra el riesgo o para la reducción de este mismo.

1. **Manual de Instalación**

Este documento sirve de guía e incluye las instrucciones para realizar de manera efectiva la instalación del producto.

1. **Material de Apoyo al Usuario Final**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento y Sistema de Ayuda en Línea, para que el usuario final pueda usar el producto exitosamente.

1. **Producto**

Los archivos del producto empaquetados y almacenadas en un CD con los conjuntos de elementos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniendo un nuevo lanzamiento al final de cada iteración, añadiendo nuevos objetivos/requisitos o mejorando los que ya fueron completados.

## Evolución del Plan de Desarrollo del Software

El Plan de desarrollo se analizará y revisará semanalmente, y será corregido al inicio de cada iteración de ser necesario.

# Organización del Proyecto

## Participantes en el Proyecto

**Jefe de Proyecto**.

Jesús Arturo Ayón León. Tiene la capacidad de mantener la comunicación entre el cliente y los usuarios, así como también mantener al equipo del proyecto enfocado en los objetivos, mantenimiento de relaciones con los agentes externos, identificación de fallos y la obtención de soluciones pertinentes que se puedan aplicar en ciertas casos, se encarga de responder ante los clientes los resultados obtenidos y de regresarle un feedback a los miembros del equipo del proyecto, experiencia en la planificación de proyectos, dirección y coordinación de los recursos necesarios en todas las fases. Habilidad para la toma de decisiones de manera responsable.

**Analista de Sistemas**.

Juan Carlos Rojas García. Experiencia con el uso de tecnologías como, JavaScript, C#, Ruby, Java, Vue.js, Kotlin y manejo de versiones con Git. Capacidad de adaptar y diseñar un producto para ayudar al equipo de trabajo a trabajar de forma más eficiente, conocimientos de UML, capacidad de analizar a detalle todos los elementos involucrados en el dominio del problema, estudia las necesidades del sistema y propone el software necesario para su aplicación.

**Analistas - Programadores**.

Luis Eduardo López Vega. Experiencia en uso de tecnologías como, Java, HTML/CSS Avanzado, JavaScript, C#, jQuery, con conocimientos básico de tecnologías como, PHP, Node.js y Python, con manejo de frameworks y herramientas de desarrollo como ASP.NET Core, Vue.js, Django, orientado a desarrollar del lado del cliente, encargado de interactuar con el usuario para brindar resultados contundentes.

José Manuel Alarid Cons Gastelum. Experiencia en uso de tecnologías como, Java, C#, Node.js, JavaScript, con conocimientos básico de tecnologías como, Kotlin, y Golang, con manejo de frameworks y herramientas de desarrollo como ASP.NET Core, React, Vue.js, Django, orientado a desarrollar soluciones de lado del servidor para brindar resultados contundentes con el mejor desempeño.

**Ingeniero de Software**.

Angilberto Caro Zavala. Capacidad de conocer el negocio del cliente y sus necesidades al igual que los problemas particulares, gran capacidad de comunicación para enseñar a los clientes a utilizar los nuevos programas. Profesional con sólidas bases metodológicas en el desarrollo de software, capaz de desempeñarse en cualquier rol que esté involucrado del proceso de desarrollo de software, con experiencia en varias tecnologías, frameworks y herramientas de diseño, ASP.NET Core, Vue.js, Java, PHP, SQL y JavaScript.

## Interfaces Externas

El sistema contará con una interfaz web que será manejada por un administrador maestro y un administrador de áreas verdes, el sistema permitirá que el administrador maestro pueda dar de alta a administradores de áreas verdes y jardineros y de igual forma pueda eliminarlos o modificarlos, crear un área verde y asignar dicha área a un jardinero, a diferencia del administrador de área verde el cual solo podrá dar de alta un área verde y asignarle un jardinero y asignar tareas, ambos administradores podrán ver el progreso que tiene las áreas verdes y su estatus.

El sistema contará con una interfaz móvil que será manejada por el jardinero, el jardinero le permitirá manejar el cuidado de las áreas verdes y revisar el progreso que tiene dicha área verde, tendrá una sección de comentarios que le puede dar al área verde dependiendo si le ocurre algo al área verde o a un árbol.

## Roles y Responsabilidades

A continuación, se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

|  |  |
| --- | --- |
| **Puesto** | **Responsabilidad** |
| Jefe de Proyecto | El jefe de proyecto establece las condiciones de trabajo, asigna los recursos, define el perfil del equipo del proyecto, gestiona las prioridades, estima tiempos y recursos necesarios para el desarrollo del sistema, coordina las iteraciones con los clientes, planifica y controla el proyecto, además debe de tener comunicación con el equipo de trabajo, gestiona los riesgos, establece un conjunto de prácticas para asegurar la integridad y calidad de los artefactos del proyecto y supervisa el establecimiento de la arquitectura del sistema. |
| Analista de Sistemas | El analista de sistemas se encarga de capturar, especificar y validar los requisitos, interactuando con el cliente y los usuarios mediante entrevistas, diseña la base de datos que utilizara el sistema, optimizando la base de datos y utilizando las técnicas requeridas para ello. Define los actores y los casos de uso al igual que la estructura del modelo de caso de uso, de igual manera es el encargado de la elaboración del modelo de análisis y diseño. Colaboración en la elaboración de pruebas funcionales. |
| Programador | El programador es el responsable de hacer el seguimiento de su propio progreso, construye prototipos, implementa las ideas del arquitecto de software. Es el encargado de documentar el código, ya que la documentación tiene como objetivo el explicar a otros programadores aquellas cosas que no resulten evidentes o claras en el código. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario. |
| Ingeniero de Software | El ingeniero de software se encarga de determinar la gestión de requisitos, gestión de configuración y cambios, documenta y demuestra soluciones desarrollando documentación, mejora las operaciones realizando análisis de sistemas, elaboración del modelo de datos, preparación de las pruebas funcionales, actualiza el conocimiento del trabajo mediante las herramientas de desarrollo y técnicas de programación. Elabora modelos de implementación y despliegue. |

# 

# Gestión del Proceso

## Estimaciones del Proyecto

Se estima un presupuesto de $100,000 pesos mexicanos para el desarrollo completo del proyecto.

## Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

### Plan de las Fases

El desarrollo se llevará en base a fases, cada fase contendrá una o mas iteraciones, tomando en cuenta que cada iteración tendrá una longitud de 2 semanas se presenta el plan de faces en la siguiente tabla.

|  |  |  |
| --- | --- | --- |
| **Fase** | **Nro.**  **Iteraciones** | **Duración** |
| Fase de Inicio | 1 | 2 semanas |
| Fase de Elaboración | 2 | 3 semanas |
| Fase de Construcción | 3 | 6 semanas |
| Fase de Transición | 1 | 2 semanas |

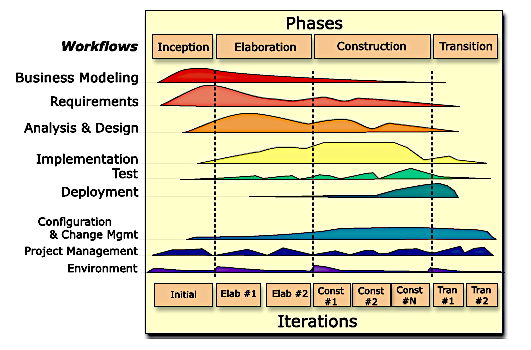
Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

|  |  |
| --- | --- |
| **Descripción** | **Hito** |
| Fase de Inicio | Esta es la primera fase, en la cual se describirán los requisitos para el sistema por parte del usuario y que serán plasmados en el artefacto Visión. Se encontrarán los casos de uso principales, y posteriormente se refinará el plan de desarrollo La aceptación de la visión y plan de desarrollo por parte del usuario indicará el final de esta fase y permitirá proseguir con la fase de Elaboración. |
| Fase de Elaboración | En esta fase se revisan los requisitos obtenidos en la fase anterior, se refinan y se genera un prototipo de la arquitectura que se usará para el resto del desarrollo. Posteriormente se analizarán los casos de uso correspondientes a los requisitos importantes que deben implementarse en el sistema. La aceptación de los prototipos de la arquitectura marca el fin de esta fase y permiten continuar a la fase de construcción. |

|  |  |
| --- | --- |
| Fase de Construcción | Se terminan de analizar y diseñar los casos de uso, al mismo tiempo se comienza con la parte de la programación del sistema, y da como resultado un release. El hito que marca el fin de esta fase es la versión de la release 2.0, la cual contendrá las funciones críticas de más importancia para el usuario, y posteriormente se entregará el producto en fase beta para iniciar pruebas con los usuarios. |
| Fase de Transición | En esta fase se prepara el release final, el cual será garantizado y preparado para su implantación en el entorno de producción, además de que se preparará al usuario para su uso.  El hito que marca el fin de esta fase incluye la entrega de la documentación del proyecto, entrega del producto y finaliza con el entrenamiento de los usuarios. |

### Calendario del Proyecto

Se presenta un calendario, el cual incluye las tareas y actividades críticas para la etapa de Inicio y Elaboración. El proceso de RUP tiene como característica realizar en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, por lo tanto, algunos artefactos se desarrollan muy pronto, pero en fases posteriores se desarrollarán aún mas, se refinarán y serán mas complejos.



Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto está lo suficientemente completo y complejo como para ser revisado y aprobado.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Disciplinas / Artefactos generados o modificados**  **durante la Fase de Inicio** | **Comienzo** | **Aprobación** | | **Modelado del Negocio** |  |  | | Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio | It #1 02/09/2019 | Aprobado | | **Requisitos** |  |  | | Glosario | It #1 04/09/2019 | Aprobado | | Visión | It #1 04/09/2019 | Aprobado | | Modelo de Casos de Uso | It #1 05/09/2019 | Aprobado | | Especificación de Casos de Uso | It #1 05/09/2019 | Aprobado | | Especificaciones Adicionales | It #1 05/09/2019 | Aprobado | | **Análisis/Diseño** |  |  | | Modelo de Análisis/Diseño | It #1 06/09/2019 | Aprobado | | Modelo de Datos | It #1 09/09/2019 | Aprobado | | **Implementación** |  |  | | Prototipos de Interfaces de Usuario | It #1 11/09/2019 | Aprobado | | Modelo de Implementación | It #1 11/09/2019 | Aprobado | | **Pruebas** |  |  | | Casos de Pruebas Funcionales | It #1 14/09/2019 | Aprobado | | **Despliegue** |  |  | | Modelo de Despliegue | It #1 14/09/2019 | Aprobado | | **Gestión de Cambios y Configuración** | Durante todo el proyecto | | | **Gestión del proyecto** |  |  | | Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones |  | Revisar en cada iteración | | **Ambiente** | Durante todo el proyecto | | |

|  |  |  |
| --- | --- | --- |
| **Disciplinas / Artefactos**  **generados o modificados durante la**  **Fase de Elaboración** | **Comienzo** | **Aprobación** |
| **Modelado del Negocio** |  |  |
| Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio | It #1 16/09/2019 | Aprobado |
| **Requisitos** |  |  |
| Glosario | It #1 17/09/2019 | Aprobado |
| Visión | It #1 17/09/2019 | Aprobado |
| Modelo de Casos de Uso | It #1 18/09/2019 | Aprobado |
| Especificación de Casos de Uso | It #1 18/09/2019 | Aprobado |
| Especificaciones Adicionales | It #1 20/09/2019 | Aprobado |
| **Análisis / Diseño** |  |  |
| Modelo de Análisis / Diseño | It #1 23/09/2019 | Revisar en cada iteración |
| Modelo de Datos | It #1 26/09/2019 | Revisar en cada iteración |
| **Implementación** |  |  |
| Prototipos de Interfaces de Usuario | It #2 30/09/2019 | Revisar en cada iteración |
| Modelo de Implementación | It #2 02/10/2019 | Revisar en cada iteración |
| **Pruebas** |  |  |
| Casos de Pruebas Funcionales | It #2 03/10/2019 | Revisar en cada iteración |
| **Despliegue** |  |  |
| Modelo de Despliegue | It #2 04/10/2019 | Revisar en cada iteración |
| **Gestión de Cambios y Configuración** | Durante todo el proyecto | |
| **Gestión del proyecto** |  |  |
| Plan de Desarrollo del Software en su versión 2.0 y planes de las Iteraciones |  | Revisar en cada iteración |
| **Ambiente** | Durante todo el proyecto | |

## Seguimiento y Control del Proyecto

**Gestión de Requisitos**

Los Requisitos a los que se les dará principal seguimiento son los siguientes:

* El sistema mostrará un diseñador gráfico para el área verde, y podrá agregar árboles.
* El sistema permitirá ingresar comentarios o anotaciones relevantes sobre cada árbol.
* El sistema guardará las coordenadas de cada árbol mediante geolocalización.

**Control de Plazos**

El proyecto será seguido y evaluado por el jefe de proyectos semanalmente de acuerdo con el calendario.

**Control de Calidad**

El control de calidad del proyecto tendrá como base el cumplimiento de los requisitos y la construcción de todos los artefactos descritos en este plan de desarrollo. También se verificará su calidad en las pruebas de software que se implementarán por parte del equipo de desarrollo y pruebas.

**Gestión de Riesgos**

A partir de la primera iteración se llevará un control de riesgos, en el que se enlistarán los riesgos que se enfrentaron en la iteración actual, como los solucionaron y un breve pronóstico de riesgos para las futuras iteraciones, semanalmente el jefe de proyectos analizará los riesgos presentes y futuros.

**Gestión de Configuración**

Se documentarán los artefactos de software, así como los releases del proyecto en sus distintas versiones y sus configuraciones, así como los cambios o modificaciones que sufran, en cada iteración se registrarán los artefactos con los que se trabajo estableciendo una versión nueva si es necesario.