**CAPÍTULO III**

**MARCO METODOLÓGICO**

**3.1 Tipo de Investigación**

El presente proyecto se ubica en la modalidad de tipo documental y factible ya que es una propuesta viable destinada a satisfacer necesidades específicas a partir de un diagnóstico. Apoyada en una investigación de campo, en el manual de Trabajos de Grado de Especialización y Maestría y Tesis Doctorales de la Universidad Pedagógica Experimental Libertador (UPEL, 2006) que se define como:

El proyecto factible consiste en la investigación, elaboración y desarrollo de una propuesta, de un modelo operativo viable para solucionar problemas, requerimientos o necesidades de organizaciones o grupos; puede referirse a la formulación de políticas, programas, tecnologías, métodos, o procesos. (p. 13).

**3.2 Diseño de la Investigación**

Como se menciona anteriormente, este proyecto se apoya en una investigación de tipo descriptivo, documental y de campo, ya que según Zorrilla (1993), expresan lo siguiente:

La investigación documental es aquella que se realiza a través de la consulta de documentos (libros, revistas, periódicos, memorias, anuarios, registros, códices, constituciones, etc.). La de campo o investigación directa es la que se efectúa en el lugar y tiempo en que ocurren los fenómenos objeto de estudio. La investigación mixta es aquella que participa de la naturaleza de la investigación documental y de la investigación de campo. (p. 43)

Dicho esto, se considera que este Proyecto Factible apoya su investigación en un diseño de campo y documental. Ya que los datos para su realización fueron obtenidos directamente del lugar donde se basó la investigación y de otros muchos de material teórico ya existente. El presente trabajo se enmarca en la modalidad de proyecto factible, pues se plantea la solución de un problema real, que presenta mejoras a proyectos ya existentes y componentes nuevos para dar solución a dicho problema.

**3.3 Nivel de la Investigación**

La presente investigación es de tipo descriptiva, ya que se miden y evalúan diversos aspectos, dimensiones y componentes de un sistema. La investigación descriptiva según la UPEL (2006)

"Es el tipo de investigación concluyente que tiene como objetivo principal la descripción de algo, generalmente las características o funciones del problema en cuestión" (p. 9)

**3.4 Fases Metodológicas**

En esta sección se exponen los pasos que se emplearon para dar repuestas a los objetivos planteados. Se especifican las herramientas usadas en cada fase de la investigación, y se presentan los pasos que amerita el desarrollo con la metodología de Desarrollo Guiado a Pruebas (TTD), llamada también Test-Driven Development, derivando de allí sus siglas en inglés, siendo una técnica iterativa de diseño e implementación de software, donde cada nueva línea de código se escribe en respuesta a una prueba fallida que el propio programador ha desarrollado. Es parte medular del desarrollo ágil de código derivado de la metodología XP, donde resulta en una práctica crítica, y de los principios del Manifiesto Ágil. Su principal objetivo es obtener código limpio que funcione.

La idea es que los requisitos sean traducidos a pruebas, de este modo, cuando las pruebas sean exitosas se garantizará que el software cumple con los requisitos que se han establecido. TDD puede ser aplicado a dos niveles, el Nivel de micro-iteraciones, el cual es guiado por pruebas unitarias escritas por el propio programador de la aplicación, se realiza mediante tres fases: Escribir y ejecutar pruebas, escritura de código, y la refactorización (Refactoring). El siguiente es el Nivel iteración o funcional, el cual es guiado por pruebas de aceptación. Estas fases se realizan de manera recursiva hasta lograr el objetivo deseado.

**Fase I: Escritura y ejecución de pruebas**

En esta fase, se describen los distintos requerimientos y pruebas unitarias a implementar para lograr un buen funcionamiento. Esas pruebas se describieron mediante la codificación de los módulos base como lo son, el módulo de control de usuarios del sistema, control de servicios, de manejo del shell, control de los programas/paquetes, control de los archivos de configuración, entre otros; siendo base primordial para el funcionamiento del sistema. Esta codificación consta de las funciones básicas de cada módulo, pero no las necesarias para su funcionamiento, tomando en cuenta así, que a la hora de hacer la verificación la prueba falle y se pueda pasar a la siguiente fase.

**Fase II: Escritura de código**

En esta fase, luego de haber descrito y verificado la falla de las pruebas, se procede a codificar todo lo necesario para que las pruebas unitarias funcionen. Esta codificación fue un poco menos rigurosa y extensa que en la anterior fase, ya que en esta solo se escribe lo necesario para que los módulos pudieran funcionar y pasar la prueba. A medida que ocurren las iteraciones el código se va perfeccionando.

**Fase III: Refactorización (Refactoring) o Reestructuración**

En esta etapa, se rastrea el código incluyendo el aplicado en la fase I, al buscar líneas duplicadas y eliminarla reestructurando el código, alterando su estructura interna pero sin cambiar su comportamiento externo, así como también se revisa que el código cumpla con ciertos principios de diseño.

**Fase IV:** **Especificación de pruebas de Aceptación**

En esta etapa, se especifican las pruebas antes que las funcionalidades sean implementadas y una vez que el código se escribe la prueba sirve como un criterio de aceptación. Esta metodología permite enumerar las dos primeras fases del ciclo coloreando en rojo aquellas especificaciones que no se cumplen y en verde las especificaciones cumplidas. Cuando se han dado los tres pasos de la especificación que ocupa, se toma la siguiente y vuelve a repetirse.