6. Desarrollo Teorico

En el desarrollo y diseño del software de simulación nos basaremos en técnicas y metodologías utilizadas para la planeación y estructuración de un plan de trabajo con el fin de hacer del proceso de programar un proceso ágil y eficiente.

Para el desarrollo de esta aplicación, se van a utilizar metodologías de desarrollo ágil de software, principalmente XP (eXtreme Programming) y SCRUM, estas metodologías han comprobado ser la mejor opción para el desarrollo de software al compararlos con metodologías tradicionales de desarrollo de software.

En las metodologías tradicionales existen diferentes fases bien definidas, que consisten en la recopilación de los requerimientos de la aplicación, análisis de los requerimientos, diseño de una solución, diseño de la aplicación, desarrollo de la codificación, pruebas, puesta en producción, y post producción.

El problema de utilizar esta metodología, radica en que las fases solo ocurren una vez durante todo el tiempo del desarrollo del proyecto, esto conlleva a que los requerimientos del producto, son recopilados en un solo instante, en donde ambas partes tanto el cliente como el equipo de desarrollo, no tienen bien identificado el problema, o tienen una idea vaga de lo que el producto debe realizar; una vez culminada esta etapa de recopilación se procede con el diseño de la aplicación, hasta que el diseño no esté terminado no se procede a la codificación, y por último se realizan las pruebas. Lo más probable es que al final del proyecto, el cliente no esté satisfecho, con la producción final del proyecto, aunque verdaderamente el producto cumpla con los requerimientos que fueron recopilados en la etapa inicial, debido a que no existe una constante entrega del funcionalidades parciales del producto al cliente, sino que hace la entrega total del proyecto al final del desarrollo de la aplicación.

Básicamente las metodologías agiles de software, como XP o Scrum, proveen una solución de este problema, haciendo ciclos mas rápidos de producción, y procurando proporcionar un contacto más frecuente con el cliente, en donde constantemente se estén entregando funcionalidades del producto, para que el cliente pueda dar su opinión, acerca del producto, Al realizar esto se habilitará la opción de realizar realimentaciones a través del tiempo, el cliente podrá descubrir verdaderamente lo que necesita que el producto realice, por lo que al final del proyecto el cliente estará satisfecho con el producto.

8.3 Control de versiones:

El control de versiones es el mecanismo para gestionar las diferentes versiones de todos los elementos de configuración que forman la línea base de un producto o la configuración del mismo.

El control de versiones provee el mecanismo de almacenamiento de cada uno de los temas que desea gestionarse. Este habilita la posibilidad de modificar, mover, borrar cada uno de los elementos. Mantener un historial de todas las acciones realizadas con cada elemento pudiendo volver a un estado anterior dentro de este historial.

Mediante este control se tiene la posibilidad de retornar la producción del software a un estado anterior, esto puede ser necesario cuando en algún momento del desarrollo, se detecte un error en el diseño y sea necesario volver a una revisión anterior en donde se originó el error.

Este control de versión es muy útil, en el desarrollo de software, ya que sirve como documentación del código, y brinda la posibilidad de volver a una versión anterior y analizar lo ocurrido.

Cada vez que se genera una versión existe la posibilidad de proveer detalles acerca de los cambios ocurridos en esta versión, de esta forma quedará documentado todo cambio que ocurra en la aplicación.

Todo este historial y todos los archivos y elementos involucrados se encuentran en un repositorio, que se encuentra ubicado en un servidor, y puede ser accedido en cualquier parte a través de internet.

Esto posibilita las futuras complementaciones del producto, los nuevos desarrolladores puedan tener acceso a este repositorio y tengan un conocimiento mas profundo de cómo fue creado el producto desde su fase inicial, como información de su evolución hasta su fase final, y puedan seguir trabajando sobre este repositorio La herramienta que utilizaremos para el control de versionamiento, es SubVersion, el cual es de libre distribución.