19-10-2018

Ricardo Ku, Guillermo Medina, Yahaira Salas

UNIVERSIDAD AUTÓNOMA DE YUCATÁN: Facultad de Matemáticas

Sistema de Control de Versiones

Primera Entrega: Documentación

# Descripción del proyecto:

El proyecto que haremos será un sistema de control de versiones distribuidos; un control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante.

En un DVCS (como Git, Mercurial, Bazaar o Darcs), los clientes no sólo descargan la última instantánea de los archivos: replican completamente el repositorio. Así, si un servidor muere, y estos sistemas estaban colaborando a través de él, cualquiera de los repositorios de los clientes puede copiarse en el servidor para restaurarlo. Cada vez que se descarga una instantánea, en realidad se hace una copia de seguridad completa de todos los datos.

# Requisitos

* Una carpeta donde se guardarán las versiones
* El formato de los archivos será .txt
* Se podrá crear o editar un proyecto
* El Sistema tendrá una interfaz gráfica
* Una pantalla con un menú de versiones
* Se tendrá acceso a versiones anteriores
* El proyecto tendrá un menú de opciones
* Tendrá un buscador para acceder de manera sencilla a versiones previas
* Se validará cada archivo antes de subirlo
* Alertas en caso de ambigüedad



Diagrama de

Sistemas de Control

Distribuidos

|  |  |  |
| --- | --- | --- |
| Entradas | Procesos | Salidas |
| Editar | Descargar archive | Archivo editado(.txt) |
| Archivo | Buscar archivo | Registro de actividades |
| Menú | Crear usuario |  |
| Contenido(modificar) | Subir archive |  |
|  | Registro de actividades |  |

# Diagrama de clases

Imagen que contiene captura de pantalla

Descripción generada con confianza alta

# Implementación de los temas de las clases en el proyecto

En nuestro proyecto nosotros implementaremos ciertos temas específicos para lograr una mejor funcionalidad en todo el desarrollo, los temas que usaremos son:

* Super clases; usaremos la clase proyecto que será nuestra clase principal y de ahí se derivarán las otras clases.
* Herencias; nuestra clase proyecto y otras clases como usuario y archivos tendrán subclases heredadas que usarán sus atributos.
* Polimorfismo; se usará para determinar distintos parámetros como tiempo, o editor.
* Interfaz; se implementará para la visualización de ventanas y alertas.