

**Plan de Configuración**

24/05/2014

**PROYECTO**



**Que Golazo!**

**Sistema de Gestión de Torneos de Fútbol**

**GRUPO**

N°3

**CURSO**

5K2

**DOCENTES**

Ing. Zohil, Julio

Ing. Liberatori, Marcelo

Ing. Jaime, Natalia

**ALUMNOS**

Allemand, Facundo leg. 58971

Herrera, Antonio leg. 57824

Pedrosa, Paula Melania leg. 58822

Rojas Amaya, M. Florencia leg. 58577

### 

**UNIVERSIDAD TECNOLÓGICA NACIONAL**

**INGENIERÍA EN SISTEMAS DE INFORMACIÓN** **PROYECTO FINAL**

|  |  |  |  |
| --- | --- | --- | --- |
| HISTORIAL DE VERSIONES | | | |
| VERSION | **FECHA** | **RESPONSABLE** | **OBSERVACION** |
| 1.0 | 20/05/2014 | Florencia Rojas | Creación Plan de Configuración |
| 1.1 | 31/05/2014 | Paula Pedrosa | Se corrigió la v 1.0, se agregó el formato de documento. |
| 1.2 | 05/06/2014 | Paula Pedrosa | Se agregó las estructura de los repositorios |
|  |  |  |  |
|  |  |  |  |

Tabla de contenido

[Introducción 2](#_Toc401928879)

[Desarrollo 2](#_Toc401928880)

[Sistema de Gestión de la Configuración 2](#_Toc401928881)

[Estructura de Repositorio de Documentación 3](#_Toc401928882)

[Estructura de Repositorio de Código 5](#_Toc401928883)

[Versionado 6](#_Toc401928884)

# Introducción

El objetivo de este documento es detallar el plan de la Gestión de Configuración (CM), para poder proporcionar una visión general de la misma. Se aborda los repositorios que utilizamos para la gestión del proyecto y la construcción del producto, y el manejo de versiones y cambios en la documentación y código.

# Desarrollo

## Sistema de Gestión de la Configuración

Herramientas a Utilizar:

* **TORTOISESVN:** es una herramienta de control de versiones
* **GOOGLE CODE:** es el repositorio que se utiliza para almacenar todas las versiones del proyecto y líneas de base del proyecto.

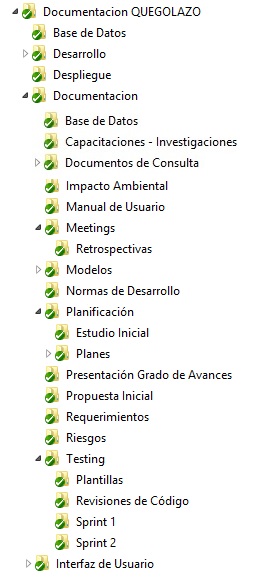
Resulta significativo aclarar que se manejan dos repositorios. Manipulamos un repositorio para el código que estamos trabajando y otro repositorio para la documentación.

Links de Repositorios:

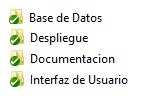
**URL Repositorio Documentación:** *https://quegolazo.googlecode.com/svn/trunk/*

**URL Repositorio Código:** *https://quegolazo-code.googlecode.com/svn/trunk/*

## Estructura de Repositorio de Documentación

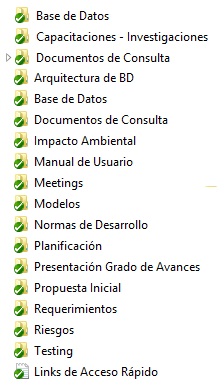


Como podemos ver en la imagen, el repositorio de Documentación está estructurado por cuatro carpetas principales. Estas son:



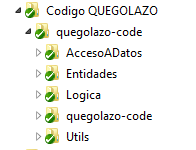
|  |  |
| --- | --- |
| **NOMBRE DE CARPETA** | **EXPLICACIÓN** |
| **BASE DE DATOS** | Esta carpeta contiene los Script de la Base de Datos de QueGolazo. Contiene 3 Script: Script de Datos, Script de Esquema y Script de Datos y Esquema. |
| **DESPLIEGUE** | Esta carpeta contendrá todo lo asociada al Despliegue del producto. |
| **DOCUMENTACIÓN** | Esta carpeta contiene todos los documentos asociados al Proyecto y al Producto que estamos desarrollando. |
| **INTERFAZ DE USUARIO** | Esta carpeta contendrá todo lo asociado a interfaz de usuario. |

Dentro de la Carpeta Documentación, se encuentran las siguientes carpetas:



|  |  |
| --- | --- |
| **NOMBRE DE CARPETA** | **EXPLICACIÓN** |
| **BASE DE DATOS** | Esta carpeta contiene todos los Documentos asociados a Base de Datos, es decir los Diagramas de Entidad - Relación |
| **CAPACITACIONES - INVESTIGACIONES** | Esta carpeta contiene todos los Documentos asociados a las Investigaciones y Capacitaciones que hemos realizado y que consideramos significativas para el desarrollo de nuestro producto. |
| **DOCUMENTOS DE CONSULTA** | Esta carpeta contiene todos los Documentos de Consulta, tales como los resultado de investigaciones, capacitaciones, etc. |
| **IMPACTO AMBIENTAL** | Esta carpeta contiene todos los Documentos asociados al análisis de Impacto Ambiental de nuestro Proyecto. |
| **MANUAL DE USUARIO** | Esta carpeta contiene el Manual de Usuario de nuestro Sistema. |
| **MEETINGS** | Esta carpeta contiene todas las reuniones documentadas. Incluye las Retrospectivas. |
| **MODELOS** | Esta carpeta contiene todos los modelos que fueron realizados. |
| **NORMAS DE DESARROLLO** | Esta carpeta contiene el documento de Normas de Desarrollo. |
| **PLANIFICACIÓN** | Esta carpeta contiene el resultado del Estudio Inicial junto con la WBS y la Calendarización y los Planes realizados, tales como el Plan de Testing, Plan de Riesgos, etc. |
| **PRESENTACIÓN DE GRADO DE AVANCES** | Esta carpeta contiene todo lo asociado a la Presentación de Grado de Avance. |
| **PROPUESTA INICIAL** | Esta carpeta contiene todo lo asociado al análisis de la Propuesta Inicial |
| **REQUERIMIENTOS** | Esta carpeta contiene todo lo asociado a la Especificación de Requerimientos de Software (ERS) y Product Backlog. |
| **RIESGOS** | Esta carpeta contiene toda la documentación asociada a Riesgos. |
| **TESTING** | Esta carpeta contiene toda la documentación asociada a Testing, como lo son los Casos de Prueba, resultado de Testing, resultados de revisiones de código, etc. |

## Estructura de Repositorio de Código



Como podemos ver en la imagen el repositorio de Código está estructurado por cinco carpetas principales. Cada una de estas carpetas representa un proyecto dentro de nuestro código de aplicación web. Cada uno de estos proyectos agrupa un conjunto de clases que se comportan de manera similar.

|  |  |
| --- | --- |
| **NOMBRE DE CARPETA** | **EXPLICACIÓN** |
| **ACCESOADATOS** | Esta carpeta contiene todas las clases asociada al acceso a datos, es decir todas aquellas clases que contemplen métodos de acceso y consulta a la base de datos.  Ejemplo: DAOTorneo: contiene todos los métodos de acceso a datos asociado a la entidad Torneo, es decir, registrarTorneo(), actualizarTorneo(). |
| **ENTIDADES** | Esta carpeta contiene todas las clases que representan Entidades de nuestro proyecto  Ejemplo: Torneo, Equipo, etc. |
| **LÓGICA** | Esta carpeta contiene todos las clases que están asociadas a la lógica de negocio  Ejemplo: GestorTorneo, GestorEquipo |
| **QUEGOLAZO-CODE** | Es el Proyecto Web, agrupa las clases de presentación.  Ejemplo: Páginas web, y archivos de configuración web. |
| **UTILS** | Esta carpeta contiene las clases que dan soporte a los gestores de la capa lógica. |

## Versionado

En cuanto al manejo de las versiones, cuando hacemos un cambio en cualquier documento, se maneja de la siguiente manera:

Versión 1.0 -> Creación del Documento

Versión 1.1 -> Cuando se agregaron cambios significativos

Si los cambios que se realizaron fueron cambios menores o estéticos sigue conservando la versión anterior.

Todos los documentos deben contar con una tabla al comienzo del mismo para gestionar el historial de versiones. En la misma se debe indicar Versión, Fecha, Responsable del Cambio y Observación.

A continuación mostramos la tabla de historia de versiones:

|  |  |  |  |
| --- | --- | --- | --- |
| HISTORIAL DE VERSIONES | | | |
| VERSION | **FECHA** | **RESPONSABLE** | **OBSERVACION** |
|  |  |  |  |
|  |  |  |  |

En cuanto a los cambios en el código, cada vez que se realiza un commit, se debe comentar el cambio realizado y ese comentario debe ser representativo de la funcionalidad agregada o modificada, de manera que cualquier persona pueda entender rápidamente lo que se realizó. Es de fundamental importancia, que cuando se realice el commit todo el código debe compilar y además, no se deben encontrar ningún tipo de error que impidiera la ejecución de la funcionalidad o afectara el trabajo anteriormente realizado.