



Generador C#

Documentación



El presente documento ha sido elaborado por Switch Software Solutions (Integralss SRL) conteniendo información confidencial, por lo que está expresamente prohibida la distribución a terceros, salvo se cuente con autorización explícita en forma escrita.

Contenido

[Propósito](#_heading=h.3znysh7) **3**

[Esquema general](#_heading=h.2et92p0) **3**

[**Descripción del proceso**](#_heading=h.1qdqzuvqpdon) **4**

[Base de datos](#_heading=h.tyjcwt) 4

[Convenciones](#_heading=h.uid6gvnjqgv9) 4

[Configuración](#_heading=h.xde68qc2qb2h) 4

[Ejecución](#_heading=h.4pfmfrhuuj3s) 5

[**Versionado**](#_heading=h.l5tke86256t6) **5**

# 

# 

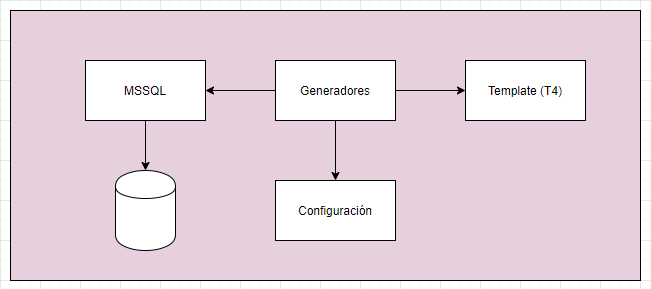
# Propósito

El desarrollo de esta herramienta tiene como propósito destinar más recursos al desarrollo de las funcionalidades particulares de cada negocio, siendo estas el core de cada sistema y generadoras de la necesidad de encomendarse en dicha tarea.

La creación automática de las operaciones CRUD de un sistema liberan al desarrollador, pudiendo enfocarse en aquellas tareas que no son (aún) posible implementarlas a través de un robot.

De esta forma, se aprovecha la capacidad intelectual de cada integrante proponiendo tareas desafiantes, sumando valor al producto y a la vez, impulsar el crecimiento profesional de cada individuo.

# Esquema general



1. El generador recibe una configuración inicial
   1. Se definen los datos necesarios para la generación del sistema. Se indican valores como nombre del proyecto, conexión a base de datos, etc.
2. A través de MSSQL, se obtiene la metadata de la base de datos sobre la que se va a trabajar.
3. Una vez obtenida la metadata, se procede a construir las clases para las operaciones CRUD, utilizando los templates.
   1. Esto se realiza reemplazando los comodines incluidos en los templates por los valores de la metadata.

# Descripción del proceso

## Base de datos

Como se describió anteriormente, el generador utiliza la metadata de una base de datos para generar el resultado (se puede decir que se utiliza un modelo de Database First).

En la versión actual del generador está disponible solamente la conexión con Microsoft SQL Server (MSSQL). Esto implica que la generación debe de realizarse a través de MSSQL, pudiendo ser la base de datos del sistema final otra cualquiera, por supuesto, con la misma estructura con la que se generó.

### Convenciones

Desde el generador se espera que se hayan respetado ciertas reglas en la construcción de la base de datos. Esto a priori no implica mayor esfuerzo al momento de la construcción como se detalla a continuación.

* Nombres de tablas
  + Las tablas deben de nombrarse en singular
    - Persona, Usuario, Proveedor y no Personas, Usuarios, Proveedores.
  + Si bien este punto no interfiere con la generación, al momento de tratar la pluralidad en código se utilizan sustantivos como PersonaList o ProveedorCollection, evitando lidiar con la pluralización del lenguaje natural (“s”, “es”, “as”, etc).
* Primaries Keys
  + Cada tabla debe de tener por lo menos una primary key.
* Foreing Keys
  + Las referencias a otras tablas deben de mantener el formato NombreCampo + Id.
    - Como ejemplo PersonaId, AutoId, etc.

## Configuración

Los valores requeridos serán:

* SaveOnDisk
  + Permite generar el código solamente en memoria
  + El código generado se presenta en pantalla y no se guarda a disco
* ProjectName
  + Nombre del proyecto que dará nombre a la solución .Net
* BasePath
  + Dirección base donde se guardará el código generado
* DatabasePassword
  + Clave de usuario de la base de datos
* DatabaseUserName
  + Nombre de usuario de la base de datos
* DatabaseName
  + Nombre de la base de datos
* DatabaseServer
  + Nombre de la instancia SQL
* DataTypeMapping
  + Dirección física del archivo de mapeo de datos SQL => .NET

## Ejecución

## 

# Versionado

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Responsable | Comentario | Versión |
| 19/10/2020 | Neri Custodio | Creación del documento | 1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |