

GCG

-

A Generic Code Generator

Vincent Hénaux

April 17, 2024

Contents

1 Introduction	2
2 Déclaration de mission	2
3 Aperçu de la solution	2
4 Cas d'utilisation	4
5 Fonctionnalités clés	5
6 Propriété intellectuelle	6
7 Avantages	7
8 Tarification et licences	8
9 Prérequis	8
10 Informations de contact	8

1 Introduction

GCG, abréviation de Generic Code Generator, est un outil polyvalent conçu pour automatiser la création de segments de code répétitifs dans les projets de développement logiciel. Il peuple dynamiquement les fichiers modèles avec des données provenant de fichiers de configuration, fonctionnant sans contraintes de langage, de framework ou de technologie. Le fichier généré par GCG est compatible avec tous les compilateurs et interpréteurs, assurant une intégration transparente dans les environnements de développement existants.

Dans le domaine du génie logiciel, le développement de code répétitifs est un aspect inévitable de la création de projets, dicté par des spécifications du langage, des exigences du framework et des design patterns. L'objectif principal de GCG est de soulager les ingénieurs logiciels du fardeau de l'écriture de ce type de code, libérant ainsi un temps précieux pour des tâches plus créatives et à plus forte valeur ajoutée.

En exploitant la puissance de GCG, les développeurs peuvent significativement améliorer leur productivité en minimisant le temps passé sur des tâches de codage fastidieuses et sujettes aux erreurs. Ces segments de code répétitifs, servent souvent de fondation aux projets logiciels, rendant leur précision et leur cohérence essentielles à la réussite du projet. GCG permet aux développeurs de renforcer leurs bases de code contre les erreurs et les incohérences, améliorant ainsi la maintenabilité et la fiabilité globales.

2 Déclaration de mission

La mission de GCG est d'aider les développeurs avec un outil polyvalent et efficace qui optimise leurs flux de travail de codage. GCG s'engage à améliorer la productivité et la qualité du code des développeurs en simplifiant les tâches répétitives, en favorisant la cohérence et en facilitant la collaboration au sein des équipes de développement.

GCG est conçu pour simplifier les tâches de codage complexes, réduire les erreurs et permettre aux développeurs de se concentrer sur des aspects plus innovants et à plus forte valeur ajoutée du développement logiciel. Notre objectif est de favoriser une culture d'efficacité et d'excellence, accélérant la livraison des projets et générant des résultats positifs dans les efforts de développement logiciel.

3 Aperçu de la solution

GCG offre une solution polyvalente pour la génération de code, disponible sous plusieurs formes adaptées à différents environnements de développement. Que vous travailliez sous Windows ou Linux, GCG fournit les outils dont vous avez besoin pour automatiser la génération de code et rationaliser vos flux de travail de développement.

Au cœur de GCG se trouve sa capacité à interpréter des fichiers modèles contenant des commandes spéciales et des espaces réservés, et à les peupler avec des données provenant du fichier de configuration. Ces commandes spéciales agissent comme des directives pour GCG, guidant le processus de génération et garantissant que la sortie répond aux spécifications souhaitées.

L'utilisation de GCG est un processus simple et intuitif. Les utilisateurs commencent par créer un template qui définit la structure et la mise en page du code à générer. Ce fichier modèle peut inclure des espaces réservés pour des données dynamiques, indiqués par des commandes spéciales reconnues par GCG.

Ensuite, les utilisateurs fournissent un fichier de configuration contenant les données nécessaires pour peupler les espaces réservés dans le fichier de modèle. Ce fichier de configuration sert de référentiel pour des informations telles que les valeurs de variables, les paramètres de fonction ou toute autre donnée nécessaire pour personnaliser le code généré.

Une fois que les fichiers modèles et de configuration sont préparés, les utilisateurs peuvent utiliser GCG sous différentes formes :

- **Exécutable:** L'exécutable GCG offre une solution autonome pour la génération de code. Les utilisateurs peuvent exécuter l'exécutable directement à partir d'un terminal, en fournissant les fichiers modèles et de configuration nécessaires en entrée pour initier le processus de génération de code.
- **Bibliothèque dynamique C:** GCG est disponible sous forme de bibliothèque dynamique pour les utilisateurs qui souhaitent intégrer ses capacités de génération de code dans leurs applications via des appels de fonction. Cette option flexible permet aux développeurs d'incorporer directement la fonctionnalité de GCG dans leurs projets, quel que soit le langage de programmation utilisé.
- **Wrapper Python pour la bibliothèque C:** GCG propose également un wrapper Python pour sa bibliothèque dynamique C, offrant un moyen facile d'appeler GCG dans des langages de script. Cette enveloppe simplifie l'intégration avec des scripts Python, permettant aux utilisateurs de tirer parti des fonctionnalités de génération de code de GCG dans leurs projets sans contraintes de langage.

Chaque version de GCG est conçue pour fonctionner de manière transparente dans son environnement respectif, garantissant la compatibilité et la facilité d'utilisation sur différentes plateformes de développement. Que vous préférerez travailler avec des exécutables, des fonctions C ou des scripts Python, GCG offre la flexibilité et les fonctionnalités dont vous avez besoin pour automatiser la génération de code et améliorer votre processus de développement.

Template file

```
USE /*{{{ VALUE database }}}*/;  
  
CREATE TABLE /*{{{ VALUE class-name }}}*/ (  
  system_id BIGINT NOT NULL AUTO_INCREMENT,  
  
  /*{{{ FOREACH member members }}}*/  
  /*{{{ VALUE member name }}}*/ /*{{{ VALUE member type }}}*/,  
  /*{{{ END }}}*/  
  
  /*{{{ FOREACH key foreign_keys }}}*/  
  FOREIGN KEY (/*{{{ VALUE key name }}}*/)  
    REFERENCES BIGINT (system_id),  
  /*{{{ END }}}*/  
  PRIMARY KEY (system_id)  
);
```

Configuration file

```
{  
  "database": "project",  
  "class_name": "user",  
  "members": [  
    {  
      "name": "email",  
      "type": "VARCHAR"  
    },  
    {  
      "name": "admin",  
      "type": "BOOL"  
    },  
    {  
      "name": "signup",  
      "type": "DATE"  
    },  
  ],  
  "foreign_keys": [  
    "team",  
    "permissions"  
  ]  
}
```

GCG



Generated file

```
USE project;  
  
CREATE TABLE user (  
  system_id BIGINT NOT NULL AUTO_INCREMENT,  
  
  email VARCHAR,  
  admin BOOL,  
  singup DATE,  
  
  FOREIGN KEY (team)  
    REFERENCES BIGINT (system_id),  
  FOREIGN KEY (permissions)  
    REFERENCES BIGINT (system_id),  
  PRIMARY KEY (system_id)  
);
```

4 Cas d'utilisation

GCG offre une large gamme de cas d'utilisation, permettant aux développeurs d'automatiser divers aspects du développement logiciel et de rationaliser leurs flux de travail. Certains cas d'utilisation clés de GCG comprennent :

- **Génération de fonctions CRUD pour l'interaction avec les bases de données:** GCG simplifie la création de fonctions CRUD (Create, Read, Update, Delete) pour l'interaction avec les bases de données. Les développeurs peuvent définir des modèles pour les opérations de base de données courantes et utiliser GCG pour générer le code correspondant, réduisant ainsi l'effort manuel et assurant la cohérence entre les projets.

- **Création d'endpoints API REST:** Avec GCG, les développeurs peuvent générer des endpoints API REST. En définissant des modèles spécifiques pour les APIs, GCG peut automatiquement générer le code nécessaire pour exposer des données et des fonctionnalités via HTTP, accélérant le développement d'applications et de services web.
- **Génération de logique de validation des données:** GCG facilite la génération de logique de validation des données pour le traitement des entrées. Les développeurs peuvent définir des modèles pour les règles de validation des données et utiliser GCG pour générer du code de validation, assurant ainsi l'intégrité et la sécurité des données dans leurs applications.
- **Automatisation des tâches répétitives dans des frameworks ou des bibliothèques spécifiques:** GCG peut automatiser les tâches répétitives associées à des frameworks ou des bibliothèques. Qu'il s'agisse de générer du code de base, de configurer des paramètres ou de mettre en œuvre des design patterns courants, les développeurs peuvent tirer parti de GCG pour automatiser les tâches routinières et se concentrer sur des aspects plus complexes du développement.
- **Génération d'interfaces entre projets et frameworks:** GCG peut générer des interfaces entre projets et frameworks, facilitant l'intégration transparente et simplifiant les mises à jour de versions de frameworks. En définissant des modèles pour les composants d'interface, les développeurs peuvent utiliser GCG pour adapter automatiquement leur base de code aux changements des exigences du framework, réduisant ainsi l'effort et le risque associés aux mises à niveau du framework.
- **Automatisation de la documentation du code:** GCG peut automatiser la génération de la documentation du code, réduisant l'effort manuel requis pour documenter les modifications de code et assurant une documentation cohérente entre les projets.
- **Génération de code personnalisés:** Au-delà des cas d'utilisation prédéfinis, les développeurs peuvent créer des workflows de génération de code personnalisés adaptés à leurs besoins spécifiques. La flexibilité et l'extensibilité de GCG permettent aux développeurs d'adapter l'outil à une large gamme de cas d'utilisation et projets.

En offrant une solution de génération de code polyvalente et personnalisable, GCG permet aux développeurs d'augmenter leur productivité, de maintenir la qualité du code et d'accélérer la livraison des projets dans divers domaines et industries.

5 Fonctionnalités clés

GCG propose un ensemble complet de fonctionnalités conçues pour rationaliser la génération de code et améliorer le travail de développement. Les principales

fonctionnalités de GCG sont :

- **Génération de code basée sur des modèles:** GCG permet aux développeurs de définir des modèles de génération de code, ce qui permet la création de structures réutilisables pour des tâches de codage courantes.
- **Personnalisation basée sur la configuration:** Les développeurs peuvent personnaliser la génération de code en fournissant des fichiers de configuration contenant les données nécessaires pour remplir les espaces réservés des modèles. Cette approche basée sur la configuration garantit la flexibilité et l'adaptabilité aux exigences du projet.
- **Commandes spéciales et espaces réservés:** GCG prend en charge des commandes spéciales et des espaces réservés dans les modèles, permettant aux développeurs d'insérer des données dynamiques et de la logique dans le code généré.
- **Compatibilité multiplateforme:** GCG est disponible pour les environnements Windows et Linux, garantissant la compatibilité sur différentes plateformes de développement.
- **Plusieurs formes de déploiement:** GCG est disponible sous différentes formes, notamment en tant qu'exécutable, une bibliothèque dynamique C, et un wrapper Python pour la bibliothèque C, offrant ainsi une flexibilité dans les options de déploiement pour répondre aux différents besoins des projets.
- **Gestion des erreurs et logs:** GCG fournit des mécanismes robustes de gestion des erreurs et de logs, permettant aux développeurs de diagnostiquer et de résoudre efficacement les problèmes pendant le processus de génération de code.

En offrant un ensemble riche de fonctionnalités et de capacités, GCG permet aux développeurs d'automatiser la génération de code, d'augmenter leur productivité et de maintenir la qualité du code à travers divers projets de développement et environnements.

6 Propriété intellectuelle

GCG s'engage à respecter et à protéger les droits de propriété intellectuelle de ses utilisateurs. Notre approche en matière de propriété intellectuelle garantit que les droits de nos clients sont respectés et qu'ils conservent la pleine propriété de leur code et de leurs actifs associés.

Le composant principal de GCG, le générateur de code lui-même, est une propriété intellectuelle protégée détenue par GCG. Cela inclut les algorithmes, la logique et les fonctionnalités qui permettent la génération de code en fonction des modèles et des configurations fournis. Cependant, il est important de noter que

GCG ne revendique pas la propriété des modèles, des fichiers de configuration ou du code généré par l'outil.

Les modèles et fichiers de configuration fournis par les clients restent leur propriété exclusive. Ces fichiers contiennent les spécifications, structures et données uniques nécessaires à la génération de code. GCG respecte la confidentialité et la propriété de ces actifs et ne conserve aucun droit sur eux.

De même, le code généré par GCG vous appartient entièrement. Qu'il s'agisse de fonctions CRUD, d'endpoints d'API REST, de logique de validation des données ou de tout autre code généré à l'aide de GCG, les clients conservent la pleine propriété et le contrôle sur la base de code résultante. GCG ne revendique aucun droit sur le code généré et n'impose aucune restriction à son utilisation, sa modification ou sa distribution.

7 Avantages

GCG est un outil puissant conçu pour rationaliser les processus de développement logiciel et améliorer la productivité des équipes d'ingénierie. Voici quelques façons clés dont GCG peut bénéficier à votre organisation :

- **Économies de temps et d'efforts:** GCG réduit considérablement le temps et les efforts consacrés à l'écriture de code redondant ou répétitif, sujet aux erreurs et pouvant être fastidieux pour les développeurs. En automatisant ces tâches, GCG permet aux ingénieurs de se concentrer sur des aspects plus innovants et à valeur ajoutée de leurs projets, augmentant ainsi la productivité globale.
- **Cohérence et maintenabilité:** Avec GCG, l'utilisation de modèles garantit la cohérence à travers l'ensemble du code source. Plus les modèles sont réutilisés, plus le code devient standardisé et plus facile à maintenir, ce qui conduit à une amélioration de la qualité du code et à une réduction de la dette technique au fil du temps.
- **Propagation de la correction des bugs:** En traitant les bugs ou en apportant des mises à jour dans un modèle, GCG garantit que chaque projet utilisant ce modèle est automatiquement corrigé ou mis à jour. Cette approche centralisée de la correction des bugs accroît l'efficacité et réduit le risque d'incohérences entre les projets.
- **Avantage concurrentiel par la propriété intellectuelle:** En engrangeant de la propriété intellectuelle à travers les modèles créés avec GCG, les entreprises peuvent acquérir un avantage concurrentiel sur le marché. Les modèles représentent des actifs précieux qui permettent aux organisations de fournir des solutions logicielles plus rapidement, moins coûteuse et de meilleure qualité à leurs clients. Cette utilisation stratégique de la propriété intellectuelle distingue les entreprises de leurs concurrents et les positionne en tant que leaders dans leurs industries respectives.

En plus de ces avantages, GCG offre des avantages distincts par rapport aux approches traditionnelles telles que l'abstraction et la méta-programmation:

- **Complexité simplifiée et maintenabilité améliorée:** Les modèles GCG maintiennent un niveau de complexité similaire à celui de simple code, les rendant plus faciles à comprendre et à maintenir par rapport aux solutions abstraites ou méta-programmées. Cette simplicité favorise une plus large diffusion des connaissances au sein des équipes de développement et garantit une meilleure maintenabilité à long terme de la base de code.
- **Processus de débogage facilité:** Avec GCG, les développeurs peuvent déboguer le code généré de manière plus efficace, car ils peuvent le lire et l'analyser comme s'il avait été écrit par quelqu'un d'autre. Cela élimine la nécessité de naviguer à travers des couches d'abstraction ou d'interpréter des constructions complexes méta-programmées, ce qui conduit à une résolution plus rapide des problèmes et à une amélioration de la qualité du code.

En exploitant la puissance de GCG et en tirant stratégiquement parti de la propriété intellectuelle à travers les modèles, les entreprises peuvent transformer leurs processus de développement logiciel, acquérir un avantage concurrentiel et fournir des solutions supérieures à leurs clients.

8 Tarification et licences

	Nombre d'utilisateurs		
	1-9	10-199	200+
Accès complet à GCG	✓	✓	✓
Maintenance	✓	✓	✓
Personnalisations simples gratuites	✗	✓	✓
Licence	Voir genericcodegenerator.com		

9 Prérequis

GCG est conçu pour une utilisation immédiate, sans prérequis ni configuration étendue requise.

10 Informations de contact

genericcodegenerator.com

genericcodegenerator@gmail.com