|  |
| --- |
| NoSQL POCO Generator  Release Notes |
|  |
|  |
| **Archita Dash** |
|  |

# Table of Contents

Contents

[Table of Contents 2](#_Toc395705959)

[Release Details 3](#_Toc395705960)

[Introduction 3](#_Toc395705961)

[System Requirement 4](#_Toc395705962)

[Features Added in the Build 4](#_Toc395705963)

[Outstanding Issues 4](#_Toc395705964)

[Known Limitation 4](#_Toc395705965)

[Contact Information 4](#_Toc395705966)

# Release Details

|  |  |
| --- | --- |
| **Product Name** | NOSQL POCO Generator |
| **Platform** | Windows 7 |
| **Date** | 13-Aug-2013 |
| **Application Version** | 1.0.0 |
| **SVN Revision Number** | 1 |

# Introduction

NoSQL Poco Generator is a simple yet productive tool that can help a developer to quickly and seamlessly use various NoSQL databases. The architecture of this application is such that based on the requirement it can integrate with Code-First as well as Model-First type of development for any NoSQL database and using any developer preferred language.

The Goal of NoSQL PO(C) JO Generator is to give a simple and powerful tool to developers using which developers can quickly integrate their application with any NoSQL database provider.

The tool is built to give high flexibility for the developer. The tool can be used in following scenarios:

* When developer wants to create models from database scripts through clipboard.
* When Model is already coded and the developer wants to create a database out of the code.
* When the database exits and developer wants to create POCO/POJO from the database by connecting to it.

This tool also provides basic syntax checking for the database code when the script is generated from clipboard. This can be used to verify the database script and at the same time look at how the model looks like when not connected to a live database. The tool is written using .NET/C# language but is written so that it gives flexibility for any language developer to use the POCO/POJO files generated from the tool.

In case an application is using Code –First Modelling approach in that case also the application can connect to the specified NoSQL Database and create the database script which can be directly executed on the database to create the repository.

For the current release it uses Cassandra as the NoSQL database and C# and VB as the language of choice. Going forward it is planned to accommodate various other NoSQL databases like Mongo DB, Couch DB, and Raven Db etc. and apart from c# and VB other language supported will include Python, C++, and Java. Currently, the user can run this application from the WPF desktop UI provided. And use various options from the Wizard kind of UI to generate database/code files and Save the code files, Create repository in the Cassandra database.

# System Requirement

1. Windows 7 desktop
2. Microsoft .Net Framework 4.5
3. Microsoft Visual Studio 2013 /2012

# Features Added in the Build

1. Database support – Cassandra using Cassandra sharp driver 3.3.0
2. Language support – C# and VB.Net
3. Support for Code First
4. Support for Model First
5. Support for Clip board based minimal syntax checking for Cassandra

# Outstanding Issues

NA

# Known Limitation

The application has not been tested end-to-end on Windows 8 Desktop.

# Contact Information

1. Seetharaman
   1. Email: [seetharaman.s@happiestminds.com](mailto:seetharaman.s@happiestminds.com)
2. Archita
   1. Email : [Archita.dash@happiestminds.com](mailto:Archita.dash@happiestminds.com)