Available Online at http://www.recentscientific.com

International Journal of Recent Scientific Research
Vol. 8, Issue, 2, pp. 15514-15515, February, 2017

ISSN: 0976-3031

International Journal of Recent Scientific Research

Research Article

CODRAW-GENERATING AUTOMATED EXECUTABLE CODE FROM UML MODELS

Purvisha Khunt., Jignesh Lad., Dhruv Tank and Reena Lokare

Department of Information Technology, KJ. Somaiya Institute of Engineering and Information Technology University of Mumbai, Maharashtra, India

ARTICLE INFO

Article History:

Received 06th November, 2015
Received in revised form 14th December, 2016
Accepted 23rd January, 2017
Published online 28th February, 2017

Key Words:

Coding skills, diminish outsourcing, fully executable, java, UML.

ABSTRACT

UML is abbreviation of Unified modelling language. Basically used for object oriented software engineering. Here the proposed system is using mainly two diagrams class diagram and activity diagram, class diagram is used for structural view of the system and activity diagram behavioural view of the system. Proposed system is mainly used for reducing outsourcing. Small applications can be easily developed without having coding skills. Proposed system will be able to convert UML diagrams into fully executable java code. In small industries proposed can be used to diminish outsourcing. Small applications based on java can be effortlessly created.

INTRODUCTION

The proposed system encourages user to generate code from UML diagram. This transformation tool generates a fully executable code from the class diagram and activity diagram, where class diagram parses structural view and activity diagram parses behavioural view. The proposed system takes xml file as an input and an output as a fully executable java code. It provides opulent interactive user interface. Proposed system is an open source. User interface of the Codraw will allow drawing UML diagrams. It provides rich user interactive interface which will allow drawing activity diagram and class diagram. This will make easy for the user to draw their problems. Generated UML diagrams can be saved as image in png or jpeg format. It allows user to refer their diagram whenever needed. After generating a UML diagram on the interface user can easily generate a fully executable java code with the help of the generate button. UML diagram parser will transform UML models into executable java code. This parser will contain an algorithm which will generate a java code in a particular file. Generate button will create a java code from UML diagram and it will store in the .exe file. This will be done by the code file creator. The .exe file will be present at a particular location in the system. It makes it easy for users to develop applications just by drawing UML models. Changes can be done by the user in the existing UML diagram whenever required. User will be able to make changes in the file which contain java code. With the help of this user can build small applications which required java skills.

Survey

Based on the study of the previous research papers and applications, this application provides more features. This application makes use of not only one but more than one UML diagram. This allows application to build more appropriate java code. Many existing system does not provide interactive interface [2]. The proposed system provides rich interactive user interface. This application allows user to save a created UML diagram as a image. So it can be referred whenever required. The proposed system allows user to generate a java code in a .exe file.

UML modelling technique is most frequently used in software design phase. This shows that UML diagram automatic generate source code.

Existing Systems

Ujector

Ujector uses three different UML diagram to generate java code. It uses class diagram, activity diagram and sequence diagram. It parses the UML diagram with the help of XMI parser and generates the source code [2].

*Corresponding author: *Purvisha Khunt*
Department of Information Technology, KJ. Somaiya Institute of Engineering and Information Technology University of Mumbai, Maharashtra, India

NCLass

Nclass is a open source tool. It uses class diagram for generating java code. It gives full c# and java support with many language specific elements [7].

Altova

Altova modelled data elements in XML schema using graphical XML Schema editor. It has XMLSpy feature which can generate java, c++ or c# code [6].

Table I: Comparison

Serial Number	Existing system	Proposed system
1	In existing systems users are not provided with the interactive interface.	Proposed system provides rich interactive user interface where user draw UML model according to their problem and get java code in a .exe file as output.
2	Existing system does not provide drag and drop feature since it does not provide rich user interactive interface.	Proposed system provides many additional features such as drag and drop.
3	Existing system does not save UML diagram as an image.	Once the UML diagram is drawn on a user interface, user can save that UML diagram as a image. It stores image in a .png or.jpg format.
4	Existing system is not an open source.	Proposed system is an open source.
5	Existing system provides complete code.	User can create small executable modules for large projects.

Proposed System

The proposed system consists of following four modules:

User interface

User interface in proposed system provides rich interactive interface to the user. It provides a canvas on which user can draw their UML diagram. To draw a UML diagram different tools are present on interface.

UML diagram parser

XMI parser starts the execution of the tool. It takes UML models as an input and generates UML model instances. Class Diagram parser class extracts information about UML class diagram from the input XMI and fills UML class diagram Meta model instance. It starts with package element which contains class and interface elements which referred in relationship like composition, generalization, aggregation etc. Activity and Sequence Diagram parser extracts information about UML activity and sequence diagrams from XMI and fills behavioural code.

Code generation algorithm

User can draw UML diagram. It creates jpeg and png file format for UML diagram. From UML diagram the transformation tool extracts the information and generates the executable code.

Modules integration

Integrate user interface, UML parser and algorithm. Transformation tool (CoDraw) generates java code from UML diagram.

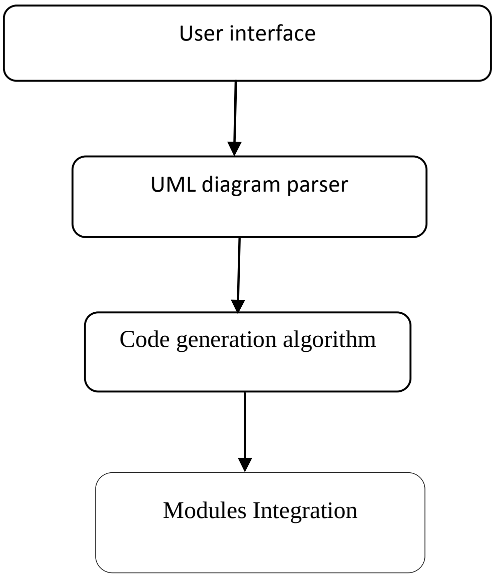

Fig. 1 System Architecture

CONCLUSION

Proposed system automatically generates fully executable java code from different UML diagrams. The objective of this proposed system is to generate efficient java code along with providing interactive user interface. It uses diagrams like activity diagram, class diagram and sequence diagram. In time to come this proposed system can be used to build more complex java application which will make use of other additional UML diagrams.

Acknowledgement

We wish to express our sincere gratitude to Ms. Reena Lokare Project Guide for providing us an opportunity to do our project work in IT domain. We sincerely thank Mr. Uday Rote, HOD of IT Department and Mr. Harsh Bhor, Project Coordinator for their guidance and encouragement in carrying out this project work. We also wish to express our gratitude to the officials and other staff members of K.J Somaiya Institute of Engineering and Information Technology.

Reference

1. B. Selic, “Uml 2: A model-driven development tool. Model driven software development,” IBM Systems *Journal, Riverton,* vol. 45, n. 3, pp. 607–620, 2006.
2. M. Usman and A. Nadeem, “Automatic generation of java code from uml diagrams using Ujector,” *International Journal of Software Engineering and its applications* (IJSEIA), Daegu, vol. 3, n. 2, pp. 21–37, 2009.
3. A. Niaz and J. Tanaka, “An Object-Oriented Approach to Generate Java Code from UML Statecharts”, *International Journal of Computer & Information Science,* vol. 6, no. 2, 2005.
4. B. Selic, “Models, software models, and uml,” UML for real: Design of embedded realtime systems, vol. Boston: Kluwer Academic Publishers, pp. 1– 16, 2003.
5. “Object Oriented UML Modeling for ATM Systems”: Department of computer technology, VJTI University, Mumbai.
6. https://www.altova.com/umodel/uml-codve-generation.html
7. http://nclass.sourceforge.net/index.html

****