ISSN:0975-9646

# Online Automated UML Generation

Avnish Singh Jat

*B.Tech Student, Jaypee Institute of Information Technology*
*A-10, Sector – 62, Noida,*
*U.P., India(201307)*

***Abstract*— This document contains a novel technique for the automated generation of UML diagrams. I have considered the automated generation of use case, sequence diagram, activity diagram and class diagram. Our technique consists of transforming text notation to UML diagram online.**

***Keywords*— UML Generation, Automated UML, Use Case, Class Diagram, Sequence Diagram and Activity Diagram.**

## I. Introduction

Unified Modeling Language plays a vital role in software lifecycle for developing the specification of the software. Designing phase of software lifecycle is about designing various diagrams required for the implementation of project.

We have tried to ease the task of developers by providing them a platform to design these diagrams by simply writing text notations. We have predefined particular syntaxes for the generation of UML diagrams.

User just needs to write the text notation and the png image of UML diagram will be generated which can be saved by the user. The proposed work can be viewed online at http://softevolve.1gh.in/uml/.

## II. Previous Works

The Unified Modeling Language was developed by Grady Booch, Ivar Jacobson and James Rumbaugh at Rational Software in 1994. In 2005 it is published by the International Organization for Standardization (ISO) as an approved ISO standard. From then much initiatives were there to increase its extensibility.

UML diagrams from text notation were initially stated by M. Fowler and C. Bock. Currently many software's have been developed for the automated generation of UML diagrams through text notation. Some of them are stated below:

1) UML Graph – It is used for automated generation of class and sequence diagram from Java-based syntax. It generates image with graphicviz diagram specification. It can be integrated in Eclipse with LightUML.
2) TextUML Toolkit - It is an open-source IDE which is recently developed that uses the UML viewer which is part of Eclipse Graphviz project to generate the diagram (currently class diagrams, ongoing work on activity diagrams).
3) yUML – It is a tool for generation of use case and class diagram online through a text notation. It is most effective tool but limited for the generation of few types of UML diagrams only.

## III. Proposed Work

The main motto behind this project is to provide a platform for developers to generate different type of UML's with text notation at a single place. We have developed a web view as well as android application which is user friendly and able to generate use case diagram, class diagram, sequence diagram and activity diagram.

For the generation of use case and class diagram we have used yuml library. The text notation entered is stored with the sessions and then pass it to preformatted tags and finally the diagram is generated with the help of Java Scripting used. In Java Script we are replacing the predefined symbols to the desired shape with the use of Java Script libraries.

### *A. Use Case Generation*

*Use case diagram is generated through predefined text notation specified below:*

1)*[Actor]:* To create an actor, the actor name is written in square brackets.

2)*(Use Case):* To create different cases '()' is used.

*3) Interactions:* Simple interaction can be generated by putting '-' between two use cases or between actor and use case. Include type interaction can be created by '<' sign and extend type interaction can be generated by '>'sign.

Example:
Input:
[Developer]-(Develop project)
(Develop project)>(Include Design)
(Develop project)<(Code)

Output:

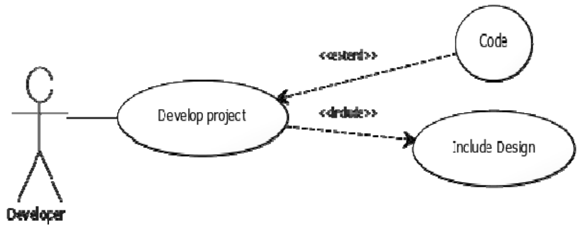

Fig. 1 Example of an automated generated use case diagram

### B. *Class Diagram Generation*

*Class diagram is generated through predefined text notation specified below:*

*1) [Class]:* To create a class, the class name is written in square brackets.

*2)* Interactions*:* For simple association ‘-’ is used, for cardinality ‘**1-0..**’, aggregation arrow is generated by ‘**+->’** , composition is shown by ‘++->’ and inheritance by ‘^-‘.

Example:
Input:
[HTML]<>-1..*[UML notation]
[UML notation]^-[Use case diagram]
[UML notation]^-[Class diagram]

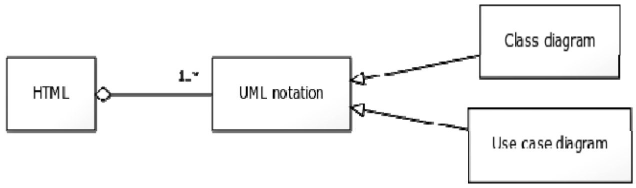

Fig. 1 Example of an automated generated class diagram

### C. *Activity Diagram*

Activity Diagram comprises of all the activities associated with project. It has a start point, set of activities, and a end point. Similarly like previous two some predefined notation is used to generate Activity diagram. Notations can be depicted from the example shown below.

Example:
Input:
(start)->|a|
|a|->|b|
|b|->(end)
Output:

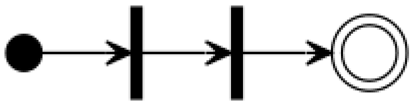

Fig. 3 Example of an automated generated activity diagram

### D. *Sequence Diagram Generation*

Sequnce diagram is drawn to show how processes interact with one another and in what order.

Input:
Title: Simple diagram
Participant B
Participant A
A->B: Message
B->A: egassem
Output:

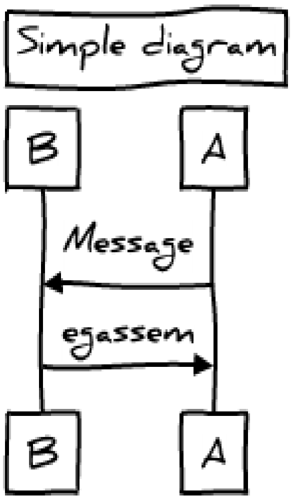

Fig. 4 Example of an automated generated Sequence diagram

## IV. Conclusions

The vital importance of UML making in software lifecycle is known to us, thus online platform is developed for users to generate the UML diagram with ease. In future, the work will extend to provide the generation of all different type of UML diagrams that are left like, package diagram, component diagram, deployment diagram, etc.

## References

[1] G Booch, J Rumbaugh, I Jacobson “Unified Modeling Language (UML)”1998.

[2] JB Warmer, AG Kleppe . The Object Constraint Language: Precise Modeling With Uml (Addison-Wesley Object Technology Series), 1998.

[3] Hans-Erik Eriksson, Magnus Penker, Astrakan, “UML toolkit”, 1997.

[4] Sinan Si Alhir, “UML in a Nutshell”, 1998.

[5] R France, A Evans, K Lano, B Rumpe, “The UML as a formal modeling notation”, 1998.