The Sustainable Efficiency of Modeling a Correspondence Undergraduate Transaction Framework by using Generic Modeling Environment (GME)

Llahm Omar Faraj Ben Dalla[1], Tarik Milod Alarbi Ahmad [2]
Software Engineering Department[1].
Atilim university, Ankara, Turkey[1], University of Gharyan[2]
Email mohmdaesed@gmail.com [1], tma_7444@yahoo.com[2]

Abstract

Nowadays, distance learning process is needed to be fully understanding for everyone all over the world. Therefore, this research study has proposed a domain specific modeling language for a correspondence undergraduate transaction framework. Furthermore, the flow of the transactions amongst diverse utilizers as well as diverse parts of a framework. Moreover, this research has defined a meta-modeling for this language via utilizing GME. Moreover, after defining the metamodel for this research study domain specific language will be automatically generated via utilizing automated tool support of the (GME). In addition, this research study will be beneficial for researchers and university transactions explanations and to simplified the educational process understandability by utilizing DSL.

Keywords: *meta-model, modeling language , GME, Generic Modeling Environment, Domain Specific language(DSL).*

1-INTRODUCTION

The paper describes how this research study has created which is associated with adapted CTS in a small modeling of DSL [1], [2] project as well as identifying the data attributes via GME according to CTS which has not studied or published before as (DSL) which is linked with some idea to simplify the framework as OS for same procedures of this kind of education [2], [3], [4]. Furthermore, this kind of education now became more readily available to the student who want to complete his education despite [6] living situation arduous life as especially with the internet age which remains associated with education via correspondence saluting that this sort of education provided to mankind OS for all segments of society which remains linked with as especially the student who working which is linked with studying. From this standpoint[7], [8], [9], [21] this research study thought modeling this sort of simplification of education as a study in order to be an initial simplification of the concept of distance education as well as development could improve in this framework as well as maintenance[10], [11], [12], [20], [22].

I. Background

A. Domain INFO

In this paper a simple case study has been presented to motivate the reader to the typical design flow developing a GME-based toolset [13], [14], [15]. Furthermore, the correspondence can study online as well as all the process will be done online, where the university frameworks should complete all the correspondence transactions. Mainly all the transactions should be done throughout the internet (related to distance education).
In addition, a correspondence can complete his education in any time that he interests in, within suitable time as well as this kind of education will be available for diverse sort of people.

B. Domain Specific Languages

A domain-specific language (DSL) is a computer language specialized to a particular application domain [16], [17], [18].
As will as it is a programming language designed specifically to come up with OSs for problems in a specific domain via utilizing GME to allow the drawing of basic diagrams for [19] a specific application domain which is associated with the coding part will be done automatically generated which can be transformed to another language, for instance, JAVA etc...

C. Metamodeling

In the first step this research has utilized GME to define the research metamodeling, which is basically a UML Class Diagram extended with some additional concepts. Furthermore, these additional concepts include defining any necessary OCL constraints as well as also some GME specific features such as configurable model visualization properties. Moreover after the meta-model remains initially defined, it can be iteratively refined until it reaches a mature state that captures all pertinent features of the domain. GME metamodels must be created utilizing the Meta-GME paradigm, which is installed which is linked with registered with GME in the same folder. however, its own metamodel can be considered the meta-metamodel. That is, it defines the concepts that are built-in to GME.all the process consists of determining two things: the entities utilized via the model, as well as the relationships amongst them. all the entities information is identifying which is linked with qualify as well as relationships will be referred to them as attributes. Metamodeling, in a nutshell, is the mapping of qualification which is linked with specification concepts onto entities, relations as well as attributes. In addition, the project name is "CTF" and is associated with the Meta-GME paradigm, the GME built-in metamodeling paradigm that configures the environment for utilizing as a meta-modeling tool.
Regarding the First step in the research meta-modeling when this study has designed meta-model initially, a correspondence domain will be done to find the basic concepts that the metamodel must contain. in GME, the metamodels identify FCOs, that is models, connections etc. Firstly, this research will build the meta-model for a correspondence flow paradigm. This study will explain the meta-model creation of a correspondence flow paradigm.

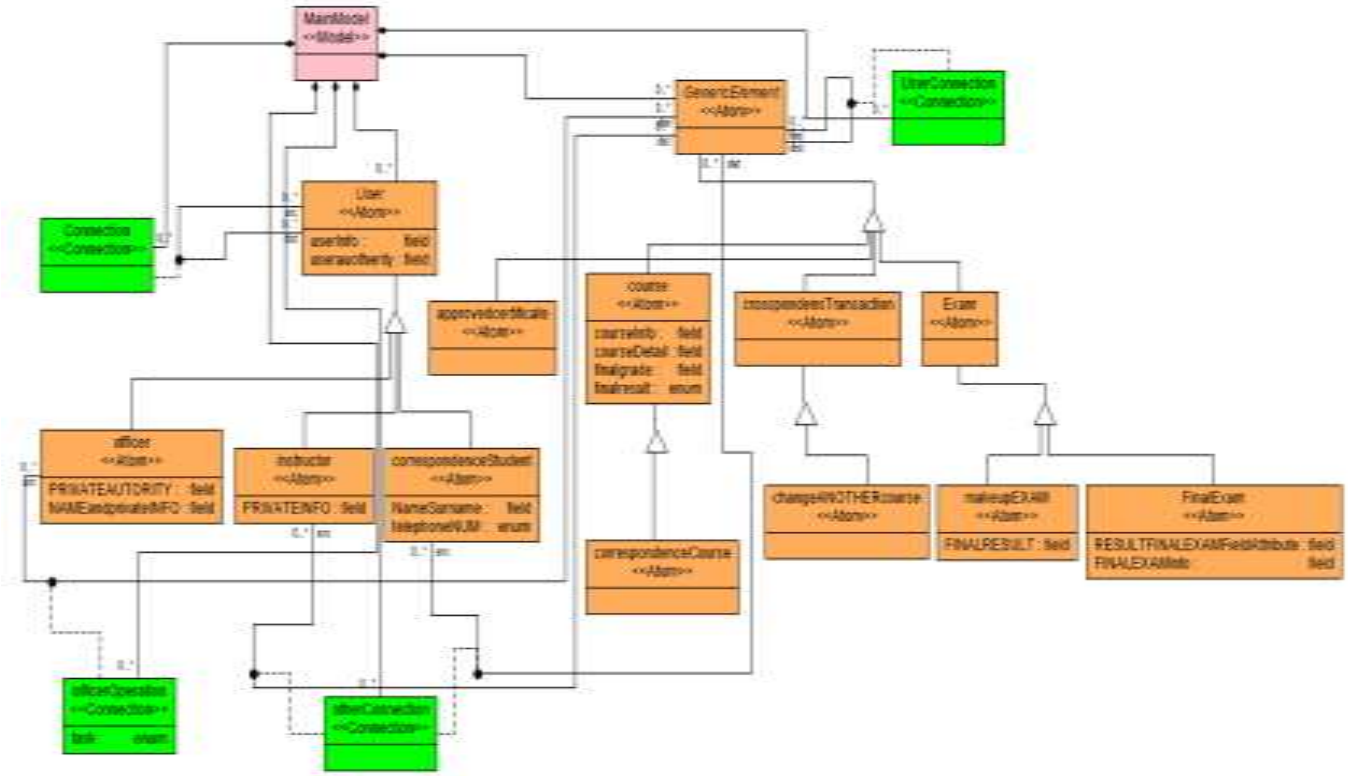


Figure.1.The proposed meta-model .

D. The research outcome of above meta-model

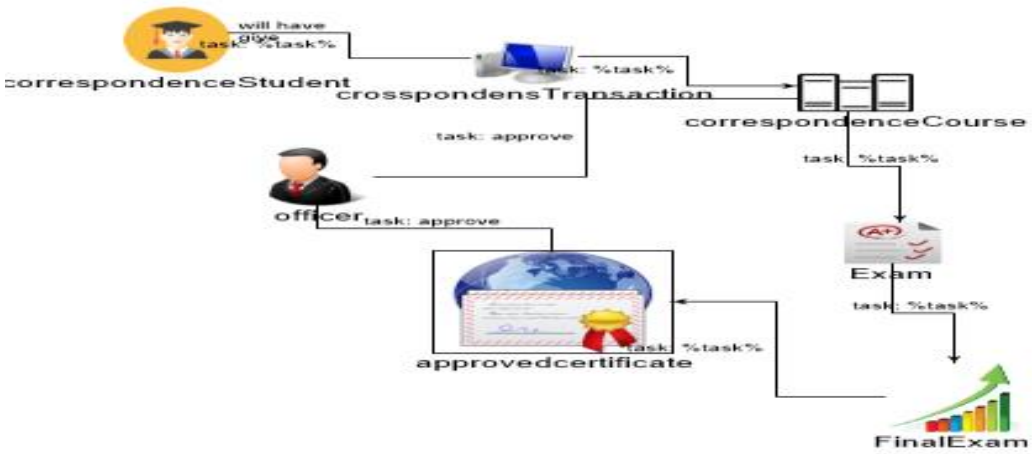


Figure .2. The first model .

This model represents a correspondence who have some transactions as well as he can ask the officer to register.
Dealings to choose his courses as well as this courses will have a result. according to this result he will be successful or fail, if a correspondence success will print the certificate as well as approved via the university. if a correspondence fail there will be two ways for him one way is makeup exam or the second way is change another course.

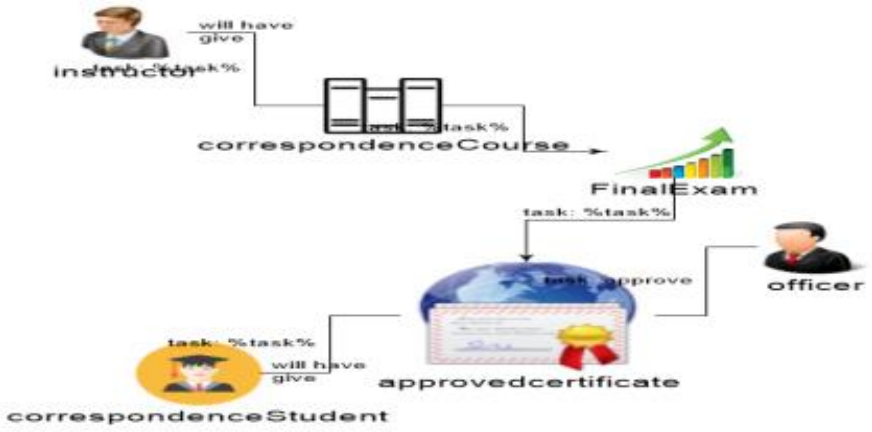


Figure.3.The second model.

This figure display instractor transaction inside the university which can be the instractor have to give acorrespondence courses as well as he will send the courses results to the officer as well as the officer will print the result which is linked with approved after that the approved certficates will be given to the a correspondence.

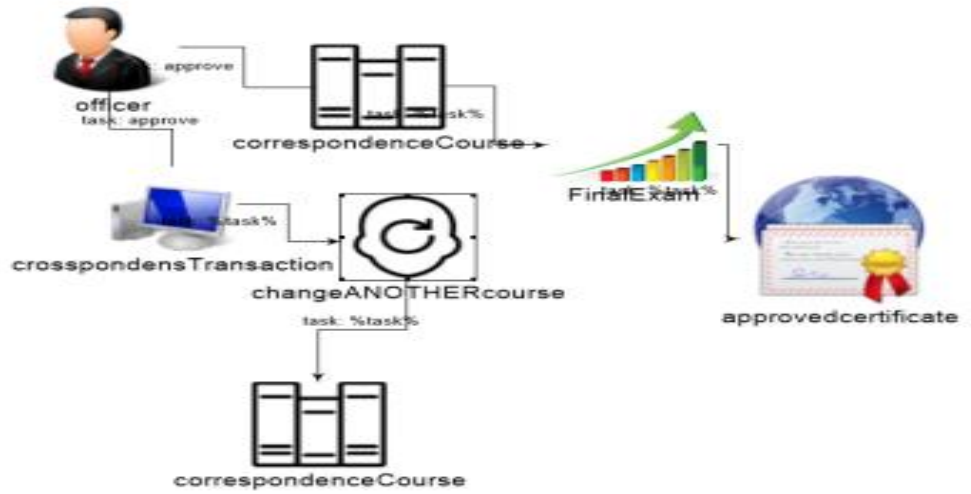
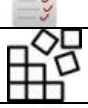
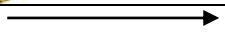
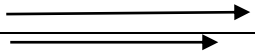
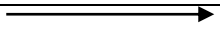
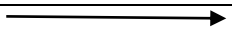
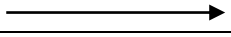
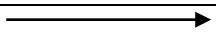
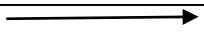
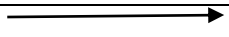


Figure.4.The third model.

I. The Meta-model for CTF

The meta-modeling of a correspondence flow paradigm in a correspondence domain.
THE ABSTRACT SYNTEX: -
the metamodel contain only one main model which named as main model also thirteen Atoms, also contain four connections as well as four connectors as well. the first term that may identify is model will be the entity that represents transaction framework. Give it the name " CTF Model" via selecting it as well as clicking the top most input field in the Attributes/Preferences/Properties window as well as change in root folder=true. then create thirteen <<Atom>> classes, as well as name them " correspondence Student "<<Atom>> which has NameSurename field attribute as well as telephoneNUM enumattribute, "user"<<Atom>>which has two field attribute userINFO as well as userauotherityField attribute, "course"<<Atom>>which has (courseINFO, courseDetail,finalgrade field attribute as well as Finalresult enum attribute), "officer"<<Atom>>which has privateAUTORITY as well as NAME which associated with privateINFO field attribute,"genericElement","approvedcertificate","crosspondenTransaction","Exam","makeupExam " <<Atom>> which has finalResult field attribute, "FinalExam" <<Atom>>which has ResultFINALEXAM field attribute as well as FinalEXAMinfo field attribute such as (date, time…..etc.),"correspondencecourse","instructor"<<Atom>> which has PRIVATEINFO field attribute. The relations will be represented via connection entities, so create a <<Connection>>.utilizer represents a user's. users could be correspondence or officer or instructor derive both the correspondence as well as the officer as well as the instructor atoms from a base user atom. I use the "Inheritance" operator (△) for this . first, connect the base class to the inheritance operator, then connect the inheritance operator to one of the objects to be derived. Thus, the same way creates correspondence courses which "Inheritance" operator (△) from course <<Atom>> .
create relationships amongst entities. Relationships are represented via lines, so switch the editor mode to "Add Connection" ( ).
This research has utilized connector (•),This is a ternary relation amongst the association class as well as the 2 endpoints of the association, so another helping object, . Only one source can be specified for each connection, however, this study can get around this via utilizing the common base class.
for example, Connect " correspondence " the connector dot. The line displays "src" as the role, as well as "0..*" to specify the destination: first click the connector dot, as well as then the " correspondence " class. The association class, "Connection", remains the third leg of the association relationship. When you connect this class to the connector dot (in any order), GME displays a window asking you to clarify the role of this relation. Select "Association Class" here.

CONCRETE SYNTAX: -

ELEMENTS AS WELL AS CONNECTIONS	NOTATION
Utilizer	
Officer	
Instructor	
correspondence Student	
Approvedcertificate	
The Course	
correspondenceCourse	
correspondenceTransaction	
changeANOTHERcourse	
Exam	
makeupEXAM	
FinalExam	
GenericElement	
UserConnection	
Connection use	
OtherConnection	
Connection use	
officerOperationConnection	
Connection use	
Connection	
Connection use	

SEMANTICS:

ELEMENTS AS WELL AS CONNECTIONS	MEANING
USER	This atom which name is user can be instructor or officer or correspondence Student [1]. In addition, the user has diverse authority according to the sort of the

	utilizer
Officer	This atom has a private field attribute which represent some transactions that the officer can be done it according to the framework need it could be print exam or print result or certificate or approve certificates as well as modify correspondence INFO .
Instructor	A tom to represent instructor as well as the relationship amongst the instructor as well as the correspondence .so the instructor has many students as well as the instructor can teach many courses during the same term .the instructor can give the exam to the correspondence as well as the result of the exam it will be send directly to the officer to be printed or to be send it to the correspondence student via the officer
correspondence Student	This atom can take some courses as well as can take exam of the course as well as for each course there is exam as well as for each exam there is a result according to the result the correspondence can be pass or can do makeup exam as well as can change to another course all these transactions will be done throughout the correspondence transaction atom
Approved-certificate	This atom represent that there will be an approved certificate will be printed or will be given to the correspondence via the officer
Course	This atom represent the general university courses which include some attributes as well as course detail to give fully courses understanding as well as the correspondence can choose his own courses from this general courses according to the correspondence degree need .

correspondenceCourse	This atom represent the course that the correspondence study it as well as this course has a result as well as has course detail which can be represent from the course atom because of the Inheritance as well as the connection to choose the correspondence courses
correspondenceTransaction	This atom represent some transactions that the correspondence can do it according to his need as well as for the term need so the correspondence can register himself as well as can choose course as well as do exam as well as get his outcome or ask for makeup exam or do make up exam or get his final result .also can ask the officer to modify his INFO
changeANOTHERcourse	This atom can represent the decision of the correspondence so he can ask the officer to change his courses .
Exam	This atom has two diverse sorts so there is FinalExam atom as well as makeupexam atom which will be done according to the correspondence result .
FinalExam	This atom represent that the correspondence can take final exam so there is field to show the Boolean attribute to show weather correspondence will pass or fail .
MakeupEXAM	Here according to the correspondence result in the final exam if he fail or get lower grade so he will do makeup exam as well as the makeup exam has final result
GenericElement	This atom made it to connect all the atoms each other
UserConnection	This is to show the connect amongst the GenericElement atom with the research mainmodel
OtherConnection	This is to show connect amongst the instructor atom as well as GenericElement atom as well as the

	research main model
officerOperationconnection	This is to show the connection amongst officer atom as well as GenericElement atom as well as connected to the research mainmodel

E. **Terminology utilized in this research study**

Terms	Full form
DSL	Domain specific language
GME	Generic modeling environment
CTS	correspondence transaction framework
PLC	project life cycle
DSM	Domain specific modeling
LOC	Line of code
MD	Meta data
PDE	project domain elements
CSE	correspondence student elements
MSL	modeling specific language
OS	optimal solution

***2*-The causes of misunderstanding the interactions of correspondence transaction framework**

- Lack of clarity in the conduct of the student transactions.
- Lack of clarification of inputs as well as outputs for this framework.
- Payment as well as guarantees, especially students who they do not have credit card or bank guarantees.
- Lack of definition of the INFO elements of the objects as well as the relationships amongst them well.

3.The benefit of modeling this kind of education.

- There is no publish of DSL for this kind of education that is why this study interest to model it for reusability to Improve CTF.

- Developing software via re-utilizing existing model to use the expert knowledge to reduce the errors as well as short time for development as well as the code will be generating automatically.
- Reduce the framework waste via extending some attributes in some classes.
- This model can be utilized again via combining the reusable part with another project.
- DMS come up with balance amongst writing the code manually as well as generating it.
- the estimation of this framework will be so effort for the programmer for the reason that the code will be generated automatically. just one analysts can perform this kind of framework.
- There will be no errors in the LOC**.**

II. Conclusion

The framework of DSM as well as a correspondence student transaction framework has increased the understanding of the DSM process throughout the researchers as well as the stakeholders as well as the framework maintainability which is linked with helped understanding the linkage amongst a correspondence student which is a associated with universities describing correspondence student elements, attributes, process. Furthermore, this study has explained throughout this project to define a correspondence metamodeling which represent in this paper an approach which can be helpful for MSL for this kind of education. To sum up DSM raises the abstraction level beyond current programming languages via specifying the OS directly via defining problem do via main concepts .usually DSM solutions are utilized in relation to a particular a correspondence student as well as normal student environment even though developers straggles amongst the OS for the problem domain , models are utilized to increase the level of abstraction which is associated with cover the implementation with high level of granularity .DSM fundamentally increase the level of abstraction as well as at the same time go down sharply with the design space despite the fact that final models are automatically generated from high level specification with domain specific code generator . In addition, this research aimed that this work will help current researchers as well as developers via being a reference as well as inspiration for future work.

III. References

1. Cai, F., He, J., Ali Zardari, Z., & Han, S. (2020). Distributed management of permission for access control model. Journal of Intelligent & Fuzzy Systems, 38(2), 1539-1548.
2. Bucchiarone, A., Cabot, J., Paige, R. F., & Pierantonio, A. (2020). Grand challenges in model-driven engineering: an analysis of the state of the research. Software and Systems Modeling, 1-9.
3. Hachem, J. E., Chiprianov, V., Babar, M. A., Khalil, T. A., & Aniorte, P. (2020). Modeling, analyzing and predicting security cascading attacks in smart buildings systems-of-systems. Journal of Systems and Software, 162, 110484.
4. Tekinerdogan, B., Babur, Ö., Cleophas, L., van den Brand, M., & Akşit, M. (2020). Introduction to model management and analytics. In Model Management and Analytics for Large Scale Systems (pp. 3-11). Academic Press.
5. Stephan, M., & Rapos, E. J. (2020). Model clone detection and its role in emergent model pattern mining: Towards using model clone detectors as emergent pattern miners–Potential and challenges. In Model Management and Analytics for Large Scale Systems (pp. 37-65). Academic Press.

6. Rademacher, F., Sorgalla, J., Wizenty, P., Sachweh, S., & Zündorf, A. (2020). Graphical and Textual Model-Driven Microservice Development. In Microservices (pp. 147-179). Springer, Cham.
7. Sebastián, G., Gallud, J. A., & Tesoriero, R. (2020). Code generation using model driven architecture: A systematic mapping study. Journal of Computer Languages, 56, 100935.
8. Ahmad, M. (2020). An analysis of educational games design frameworks from software engineering perspective. Journal of Information and Communication Technology, 14, 123-151.
9. Elrakaiby, Y., Spoletini, P., & Nuseibeh, B. (2020). Optimal by Design: Model-Driven Synthesis of Adaptation Strategies for Autonomous Systems. arXiv preprint arXiv:2001.08525.
10. Moradi, H., Zamani, B., & Zamanifar, K. (2020). CaaSSET: A Framework for Model-Driven Development of Context as a Service. Future Generation Computer Systems, 105, 61-95.
11. Hili, N., Bagherzadeh, M., Jahed, K., & Dingel, J. (2020). A model-based architecture for interactive run-time monitoring. Software and Systems Modeling, 1-23.
12. Sebastián, G., Gallud, J. A., & Tesoriero, R. (2020). Code generation using model driven architecture: A systematic mapping study. Journal of Computer Languages, 56, 100935.
13. Beebe, N. H. (2020). A Bibliography of Publications about the Java Programming Language, 2010–2019.
14. Gerl, A. (2020). Modelling of a Privacy Language and Efficient Policy-based De-identification.
15. Pérez-Castillo, R., Ruiz, F., & Piattini, M. (2020). A decision-making support system for Enterprise Architecture Modelling. Decision Support Systems, 113249.
16. Pourali, P. (2020). A User-Centric Approach to Improve the Quality of UML-like Modelling Tools and Reduce the Efforts of Modelling.
17. Camilli, M., Gargantini, A., & Scandurra, P. (2020). Model-based hypothesis testing of uncertain software systems. Software Testing, Verification and Reliability, e1730.
18. Zhu, Z., Lei, Y., Li, Q., & Zhu, Y. (2020). Exploring MDE techniques for engineering simulation models. Wireless Networks, 1-12.
19. Mustafee, N., Bae, K. H. G., Lazarova-Molnar, S., Rabe, M., Szabo, C., Haas, P., & Son, Y. J. INFRASTRUCTURE FOR MODEL BASED ANALYTICS FOR MANUFACTURING.
20. Coe, C., Piggott, C., Davis, A., Hall, M., Goodell, K., Joo, P., & South-Paul, J. (2020). Leadership Pathways in Academic Family Medicine: Focus on Underrepresented Minorities and Women. Family Medicine.
21. Han, Z., Wu, J., Huang, C., Huang, Q., & Zhao, M. (2020). A review on sentiment discovery and analysis of educational big-data. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 10(1), e1328.
22. Le, D. M., Dang, D. H., & Nguyen, V. H. (2020). Generative software module development for domain-driven design with annotation-based domain specific language. Information and Software Technology, 120, 106239.

.