Journal of Positive Sciences (JPS) **eISSN: 2582-9351, https://positive-sciences.com** **JPS. 2021 Mar.; 1(1): 13-17**

Article info **Research Article**
Received on: 19.02.2021
Accepted on: 27.03.2021
Published on: 31.03.2021
doi: https://doi.org/10.52688/ASP24661

Domain specific modeling language for agile software development types

Gaffalla Aborisha[1,*], Doğu Çağdaş ATİLLA [2], Çağatay AYDI [3]

[1, 2, 3] Department of Information Technology, ALTINBAS University, Istanbul, Turkey

[*] gaffalaaborisha@yahoo.com

ABSTRACT

This paper introduces the drive of domain specific modeling language for agile software development (ASD). The flow of the relations among various agile sorts and various inherited children of agile framework. This study motivation expresses a meta model for field of specific language via utilizing Generic Modeling Environment (GME). Subsequently the essential part remains towards producing the main meta model for this field of specific language the inherited code will be mechanically generated via utilizing automated implement sustenance of the Generic Modeling Environment.

Keywords: Meta model, modeling, generic modeling environment, domain specific language, agile software development (ASD)

INTRODUCTION

Analysts might be gone into an electronic database framework searching for some data about ASD [11]. On the other hand, legacy might be written via hand on preprinted frames that are collected into cushions, or imprinted onto comparable structures utilizing a PC printer. Now, a legacy might be transmitted from the scientists to the understudies orally via phone, although this training may expand the risk of ASD legacy [11]. The substance of an incorporates presenting the hierarchy of agile software development via way of ASD sorts name of the supplier Unique for separately lithe sort data detail remains to the name of the agile sort [22]. ASD sorts name and data detail should likewise be recorded. The paper agile framework, that was being utilized sometime recently, squandered the season of all gatherings engaged with the data detail and legacy of dexterous mother iota detail.

FILED OF MODELING A LANGUAGE

This study, existing a basic contextual analysis to acquaint the inspect with the run of the mill configuration stream building up a GME-based toolset [1]. The space of agile software development remains composed to many agile software development sorts. Also, the subdivision sorts have many various sorts such via way of lean software development and crystal family. The researcher goes directly to the library, where the agile framework can be found and explained to everyone and make sure via way of systematic researchers and then tendered over to the researcher. Whether agile sorts already have an inheritance relationship or no inheritance relationship, the online model can bounce the researchers the optimal solution to see agile sorts and their inheritance relationships suggestions to extra high-quality and suppleness with managing and receiving the data necessity. When agile sorts presenting on the utilized framework, he or she does so via electronic means, with the support of an operational form. The utilized framework stores incoming new-fangled various agile sorts and sends partitions' data on demand to agile data framework. The new-fangled addition remains then proximately reachable in every site on invitation. The venture will make the agile framework naturally approachable. Since the data communications are digitized, it significantly decreases paper practice throughout the e-learn segment to recover the E-learning and web education [12].

Partitions spend significantly less time issuing to present the optimal agile sorts relationships. They will get feedback via way of to whether the normalsearching on the digital library but it remains not enough to know all agile sorts. In terms of agile sorts, it remains very important for researchers to know whether their agile sort data has inherited from any agile sorts that has inherited to them.

***Corresponding author**
Gaffalla Aborisha,
Department of Information Technology, ALTINBAS University, Istanbul, Turkey
e-mail: gaffalaaborisha@yahoo.com

Researchers will invest fundamentally less energy filling the papers and can consider the coordinated sorts connections. For them, things are disentangled via the way that greatest of agile programming improvement data on the presenting remains via way of of now went into the framework, which remains the reason they are just required to add to the ASD data real conveyed models and data. Much of the time, this should be possible utilizing a standardized identification peruse. The medicine remains then prepared for e- invoicing to the Education Insurance Fund [22].

Specialists never again necessity to stress over conveying paper subtle foundations or losing the data. Additional real favorable position of the framework remains that scientist's visits are never again required for routine refills. Spry with the versatile to change requiring additional expansion for their consistent change would just necessity to contact their schoolmate via email or telephone, and the specialists can matter refills with a few ticks of the mouse. Afterward that, the researcher can have the agile data provided that they have the agile sorts name and demonstration their individual data

FILELD OF FACTOR SPECIFIC LANGUAGES

A field of (DSL) is a PC language expert to application factor [8,14]. This contrasts with the utilize of GPL, which is largely applicable across factors. There is a broad range over DSLs, ranging beside extensively utilized languages because of frequent factors, such namely HTML because of the site page, under according to languages utilized via solely some then a temperate piece about software, certain namely Emacs Lisp due to GNU Emacs or XEmacs [12]. DSL can remain further subdivided via the variety of language, yet consist of field-specific markup languages, specific model languages (additional generally, specification languages), yet factor of specific languages. Special-drive PC languages hold continually occurred among the PC age, however, the time " factor of specific language" has to turn out to be greater well-known due according to the upward push on factor specific modeling [12,13]. Simpler DSL, especially ones utilized via a singular application, are every now and then informally known via way of small -languages [14]. The line among general- drive languages yet factor specific languages remains now not constantly utilized to specified communication with some systems and specific method, additionally bear specialized services for a unique factor, however, be relevant additional broadly, yet conversely may also among principle successful concerning huge utility but within object utilized chiefly for a specific factor. For instance, Perl utilized to be first raised namely a text-meting out then contain language, for the equal factor via way of like AWK then case scripts, however, utilized to be and largely utilized via way of a general- drive programming call other on. via contrast, PostScript remains a Turing perfect language, and of precept execute lie utilized because of somebody task, however among act remains barely utilized via way of like a page agile sound [1,2].

Table 1. DSL for agile software development and Concrete syntax

Agile sorts	factor specific language DSL
Agile	
Lean	
Kanban	
Lean integration	
Lean manufacturing	
Crystal family	
DSDM	
FDD	

Orange crystal	
Scrum	
Lean software	
Lean startup	
XP Programming	
Red crystal	
Yellow crystal	
Adaptive agile software development	
Clear crystal	
AgileConnection	
otherConnection	

To outline a language, certain requirements a word in accordance with writing the assignment in. The speech regarding a model remains repeatedly referred to via way of a metamodel, for this reason, the speech because significant a demonstrating atom remains a meta (meta) model [1,2,4]. The meta (meta) frame do remain vindicated among pair groups: those to that amount are derive out of yet customizations regarding odd languages, and these that have been flourished especially via way of pathway concerning meta meta models. Derived meta-metamodels consist of ERD Figures, the languages will be (Formal, EBNF, Ontology, XML, MOF) [3]. The strong point of these languages tends to stand among the knowledge and adjustment regarding a unique sound [3,2,5].

*Corresponding author
Gaffalla Aborisha,
Department of Information Technology, ALTINBAS University, Istanbul, Turkey
e-mail: gaffalaaborisha@yahoo.com

Stability toughness longevity toughness permanency greatest odd DSL takes location along issue -specific language environments, either advertisements certain with the aid of pathway of Meta Edit+ yet ACTi source, originate sources certain via road about GEMS, then academic certain with the aid of road on (GME). The growing acceptance of DSL consumes carried after DSL frameworks existence extra in conformity with cutting-edge IDEs, e.g. (EMP), and Microsoft's DSL Tools for programming Factories [7]. The terminology concerning aspect -specific languages applied via utilizing a range of GME setups relies upon regarding an association about gentle ideas incorporated with GME itself [3,4,7]. The choice of this non-exclusive thoughts remains the near fundamental graph choice. GME bolsters a number thoughts because of structuring outstretched scale, complex fashions [14]. These comprise pecking order, more than a few perspectives, sets, references, yet unambiguous limitations. GME remains to an area demonstrating condition that can be designed and adjusted from meta level worldview details. In this manner, in view of the worldview, GME can be adjusted rapidly to an area device that speaks to a specific building space [14,15].

SEMANTICS

Table 2. the semantic of the Metamodel

ELEMENTS AND CONNECTIONS	MEANING
Agile	This atom which name remains Agile can be agile software development and their sorts. Also, the agile has various sorts of the agile model
Scrum	This atom has a private field attribute which represent some INFO that can be added to the agile INFO to the system need it could be print to the researcher as soon as possible.
Lean	This atom represents Lean and the relationship between the agile and the Lean. So, Lean has many childs and Lean can be a mother of other Lean sorts. Lean can give some attributes to the lean children.
Kanban	This atom can be one of the agile sorts and it is inherited from agile atom

METAMODELING

The principal's necessity does utilizing GME remains characterize an outline of the main meta model, which remains fundamentally a UML Class Figure reached out with some extra ideas [10,11,12, 13]. These extra ideas incorporate characterizing any vital OCL imperatives and furthermore, some GME highlights, for instance, configurable model representation properties. Afterward the meta model remains at first characterized and implemented, it container be iteratively refined until the point that it achieves a develop rapid that catches every applicable element of the area. This refinement brings about a change to space demonstrating dialect (DSML). Via way of the nature of (DSML) enhances, single container rapid improved space frames utilizing DSML. GME meta model's necessity be formed utilizing Meta (GME) paradigm, which remains connected and itemized with GME. Meta GME remains just additional modeling language; though, its individual meta model identified to be measured the meta meta model. That remains, it describes the perceptions that remain integral to GME. Atom: - utilized towards characterize atomic essentials terms. Model: - utilized towards characterize container essentials terms. Reference: - utilized via way of a pointer towards other essentials terms. Set: - utilized towards cluster of essentials terms. Connection: - utilized towards assistant essentials terms. These essentials terms are termed FCOs as an aspect of GME. FCOs identified to be contain both textual characteristics (List of numbers, one, zero and Field sort) and Constraints, which are OCLbased expressions for if verifiability models. Additional vital perception in GME remains to Aspect. Models can have numerous aspects that choice a subset of the modeling perceptions towards demonstrate towards the modeler at once.

Such as, a model of a distributed programming framework life cycle might have a data flow and controller flow aspect. The greatest fundamental advance in a meta modeling method comprises of determining binary essentials terms: the essentials terms utilized via the model, and the relations among them. Data utilized towards distinguish and succeed certain substances and connections will be doled out towards them via way of properties. The Meta modeling, basically, remains recording of detail ideas onto essentials terms, connections, and traits. The model remains termed "coordinated" and remains related towards the paradigm

*Corresponding author
Gaffalla Aborisha,
Department of Information Technology, ALTINBAS University, Istanbul, Turkey
e-mail: gaffalaaborisha@yahoo.com

Meta modeling, the GME worked in the main meta model paradigm that arranges the area for utilizing via way of a main metamodel apparatus.

METAMODELING FOR AGILE SOFTWARE DEVELOPMENT

At the point when this investigation began towards plan meta model at first, agile programming improvement space necessity be dissected towards locate the essential ideas that the meta model necessity contain. To begin with, this study decided to assemble the meta model for agile programming development (ASD) stream worldview. The metamodel formation of the agile programming development sorts stream worldview. In agile programming development, the meta model contain one model and atoms, also it includes connection and connector the first term that may identify remains model resolve the atom that signifies agile methods. Specifies it their names are "agile Model" via choosing it and ticking the top greatest internal field in Features / Preferences / Properties gap and change in root folder=true.

Form eight <<Atoms>> classes, "scrum", "lean", "crystal family", "FDD","DMDS", "XP Programming", "UP" “Kanban". The relationships drive represented via connection entities, so form a <<Connection>> [23,24,25,26].

Lean represents a lean family. Lean family could be LSD or lean startup or lean manufacturing derive both the lean programming development and the lean startup and the lean manufacturing atoms from lean atom. This study utilizes the “Inheritance” machinist ( ). First, linked to the class towards the inheritance operative, then linked to the inheritance machinist to one of the items towards be derived. Consequently, the same way forms crystal family and the other agile children.

Form relations among objects. Relationships are represented via lines, so switch the editor mode towards "Connection" ( ). This study utilize connector ( ), This remains a ternary relations among the association class and the two endpoints of the association, so additional serving object .One source can be specified for separately connection, then it can come to be around this via utilizing the common class. Such as, linked to "Scrum" connector ( ). The line presents " src " via way of the role, and "0..*" towards specify the destination: first click connector ( ), and then the "agile" class [24,25,26]. The association class, "Connection", remains to the third choice of the association relationship. Once the designer linked to this class towards connector ( ), GME presents a window enquiring the designer towards clarify the role of this relation. Select "Association Class" as presented in figure 1.

Afterward the completion of the design meta model form Model via converting the meta model into a GME paradigm. Via clicking on the button with the cogwheel icon (meta (GME) Interpret) in toolbar. A meta model should first be translated through the Meta (GME) mediator, and the formed paradigm ASD (an XML document which has an " .xmp " expansion) necessity be enrolled in the GME registry before in the least models of that worldview can be assembled. In the wake of doing both means, make a model of ASD. The initial phase in the structure of this framework remains towards make additional venture in GME whose meta model remains to our Prescription worldview, and afterward towards embed an App component into the Root Folder (the superlative stage of the compartment) via way of appeared in Figure 1 below.

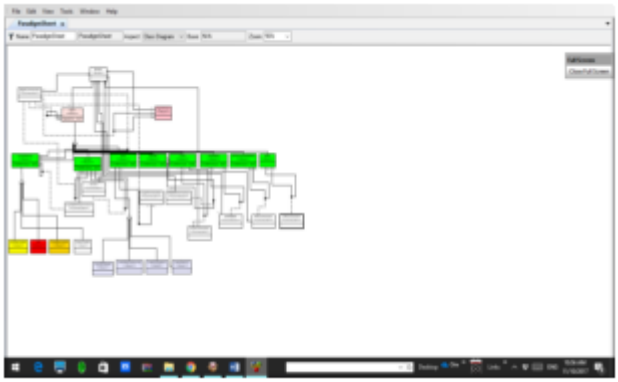

Fig. 1. Agile software development Metamodel.

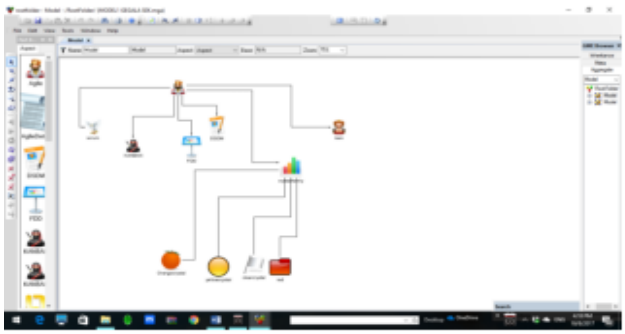

Fig. 2. Model1 of agile model.

*Corresponding author
Gaffalla Aborisha,
Department of Information Technology, ALTINBAS University, Istanbul, Turkey
e-mail: gaffalaaborisha@yahoo.com

CONCLUSION

This study presented a factor specific language of agile programming development applications through GME, which can be helpful for education framework and can give a good performance for separately generous of agile sorts and demonstration the inheritance of agile software development. Above and beyond this framework can be re-utilized distorted to any framework via way of well via way of the production and sustainable development. In (DSM), language contrives have qualified controller concluded languages, and would be able towards settle on a proper model joining approach for their circumstance. In some cases, it can be conceivable towards coordinate a few regions of interests, e.g. persistence, route, format, data, into a solitary presenting language, though at various circumstances the utilize of various languages and unequivocal mix among the models are favored. Regardless, there drive dependably be various model outlines towards incorporate, whether via components having subfigures, or components being reused or referenced in a few diagrams.

REFERENCES

[1] Habibzadeh, H.; Nussbaum, B.H.; Anjomshoa, F.; Kantarci, B.; Soyata, T. A survey on cybersecurity, data privacy, and policy issues in cyber-physical system deployments in smart cities. Sustain. Cities Soc. 2019, 50, 101660.
[2] Gartner. Gartner's Top 10 Security Predictions 2016. Available online: https://gtnr.it/2vRorda (accessed on 1 February 2020).
[3] Geer, D. The Internet of Things: Top five threats to IoT devices.Available online: https://bit.ly/2UVixkF (accessed on 1 February 2020).
[4] Ande, R.; Adebisi, B.; Hammoudeh, M.; Saleem, J. Internet of Things: Evolution and technologies from a security perspective. Sustain. Cities Soc. 2019, 54, 101728, doi:10.1016/j.scs.2019.101728.
[5] Riahi, A.; Challal, Y.; Natalizio, E.; Chtourou, Z.; Bouabdallah, A. A Systemic Approach for IoT Security. In Proceedings of the 2013 IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), Cambridge, MA, USA, 20–23 May 2013.
[6] Jesus Pacheco, S.H. IoT Security Framework for Smart Cyber Infrastructures. In Proceedings of the IEEE International Workshops on Foundations and Applications of Self* Systems, Augsburg, Germany, 12–16 September 2016.
[7] Dorri, A.; Kanhere, S.S.; Jurdak, R.; Gauravaram, P. Blockchain for IoT security and privacy: The case study of a smart home. In Proceedings of the 2017 IEEE International Conference on Pervasive Computing andCommunications Workshops (PerCom Workshops), Kona, HI, USA, 13–17 March 2017.
[8] Yao, X.; Han, X.; Du, X.; Zhou, X. A Lightweight Multicast Authentication Mechanism for Small Scale IoT Applications. IEEE Sens. 2013, 13, 3693–3701. [CrossRef]
[9] Butun, I.; Morgera, S.D.; Sankar, R. A Survey of Intrusion Detection Systems in Wireless Sensor Networks. IEEE Commun. Surv. Tutor. 2014, 16, 266–282.
[10] Debar, H.; Dacier, M.; Wespi, A. Towards a taxonomy of intrusion-detection systems. Comput. Netw. 1999, 31, 805–822.
[11] Meng, G.; Liu, Y.; Zhang, J.; Pokluda, A.; Boutaba, R.Collaborative Security: A Survey and Taxonomy. ACM Comput. Surv. 2015, 48, 1:1– 1:42.
[12] Midi, D.; Rullo, A.; Mudgerikar, A.; Bertino, E. Kalis—A System for Knowledge-Driven Adaptable Intrusion Detection for the Internet of Things. In Proceedings of the IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, USA, 5–8 June 2017.
[13] Haddad Pajouh, H.; Javidan, R.; Khayami, R.; Ali, D.; Choo, K. A Two-layer Dimension Reduction and Two-tier Classification Model for Anomaly-Based Intrusion Detection in IoT Backbone Networks. IEEE Trans. Emerg. Top. Comput. 2016, 7, 314–323.

*Corresponding author
Gaffalla Aborisha,
Department of Information Technology, ALTINBAS University, Istanbul, Turkey
e-mail: gaffalaaborisha@yahoo.com