

# Graphical User Interfaces Generation from BPMN (Business Process Model and Notation) via IFML (Interaction Flow Modeling Language) up to PSM (Platform Specific Model) Level

Abir Sajji<sup>1</sup>, Yassine Rhazali<sup>2</sup>, Youssef Hadi<sup>3</sup>

Computer Research Laboratory-Faculty of Science, Ibn Tofail University, Kenitra, Morocco<sup>1,3</sup>  
Information and Communication Systems Engineering Research Group, Higher School of Technology, Meknes, Morocco<sup>2</sup>

**Abstract**—The fundamental concept behind the MDA (Model Driven Architecture) approach is the development of many models, first the Computation Independent Model (CIM), then the Platform Independent Model (PIM), and lastly the Platform Specific Model (PSM) for the concrete implementation of the system. Web applications are just one example of customized software that is now being developed at an increasing rate. Interaction Flow Modeling Language (IFML) was developed to represent the front end of any program that necessitates a powerful interaction with a user through the use of an interface, regardless of the technical details of its implementation. There are various modeling tools for IFML; the Webratio tool is an illustration that facilitates the generation of the entire web application. This article discusses the model transformations in the MDA's approach, starting from the CIM level up to the PSM level through the PIM level. To begin, we created the Business Process Model and Notation (BPMN) and IFML metamodels in Eclipse tool, we created also the BPMN model, and we get the IFML model by applying the shift rules in Atlas Transformation Language (ATL). Finally, we generated the application using a standard tool that implements IFML Webratio tool. A CRUD (Create, Read, Update, and Delete) features for the after-sales service case study were provided to illustrate the conversion strategy from the CIM level via the PIM level to the PSM level.

**Keywords**—MDA (Model Driven Architecture); CIM (Computation Independent Model); PIM (Platform Independent Model); PSM (Platform Specific Model); Model transformations; Graphical User Interfaces; BPMN (Business Process Model and Notation); IFML (Interaction Flow Modeling Language); Webratio tool

## I. INTRODUCTION

The core idea behind the MDA's methodology is the use of models at various points in an application's development lifecycle. MDA promotes requirements development (at the CIM level), design and analysis (at the PIM level), and eventually code (PSM level). MDA's primary goal is to create models that are independent of technological aspects to generate all application code automatically and significantly boost productivity [1].

It has been increasingly crucial to deliver tools and processes for web application development. Because of how complicated are the interfaces, the increased demand for Web

applications, and businesses' overall need to create applications quickly in recent years.

Model-driven architecture's rise demonstrates its value to engineering and its effectiveness in raising application quality, while reducing development time and boosting productivity. [2].

Recently, a new web engineering technology named Interaction Flow Modeling Language (IFML) was suggested for use in developing web and mobile applications [3]. It is an Object Management Group (OMG) standard that makes it easier to describe graphical User Interfaces for desktop computers, laptops, mobile phones, and tablets without regard to the platform on which they are used.

Several tools have been suggested for creating IFML models and codes, including the Webratio tool [4]; a model-driven project with code generators that can create functional applications from IFML models.

Our main goal of this article is to automatically generate graphical user interfaces from PIM to PSM using the Webratio tool. In this article, we are going to treat the model transformations in the MDA's approach, starting early from the CIM to PIM up to the PSM level.

We created in the first step the BPMN and IFML metamodels at Eclipse tool, the business process model of a case study named the after-sales service using the BPMN notation created also, then applied the transformation rules in ATL language [5]. We obtain the IFML model at the PIM level into Eclipse tool, finally, we import the IFML model in .xmi format [32] into the Webratio tool to get graphical user interfaces (GUI) code.

The key benefit of our work is that it uses two standards BPMN and IFML as well as a powerful commercial tool Webratio. To cover every step of the model transformations process in the MDA's approach, starting from the CIM level through the PIM level and up to the PSM level.

The article is structured as follows: in Section II we go into the fundamental techniques that underpin our strategy in the theoretical background, Section III presents the related works, in Section IV we will present our approach by describing the

CIM level to the PIM level transformation rules also the representation of the two metamodels BPMN and IFML. Section V provides a case study of the after-sales service illustrating our methodology starting from the CIM level to the PIM level up to the PSM level, a discussion and limitations are presented in Section VI, and we finish by outlining the ongoing and future work in Section VII.

## II. THEORETICAL BACKGROUND

### A. BPMN

The Object Management Group (OMG) established the Business Process Model and Notation (BPMN), a standard that gives businesses the ability to graphically describe and analyze their internal business processes and consistently communicate these processes. Additionally, the graphical notation makes it easier to understand how business transactions and collaborations across companies are performed [13]. A BPMN model does not accurately depict the system's functional behavior. BPMN's fundamental conceptual primitives are tasks, events, gateways, pools, lanes, and flows [14].

### B. IFML

The OMG developed the Interaction Flow Modeling Language (IFML), to define interaction flow models. These models are employed to explain user-application interaction on the front end.

IFML was created by developers with more than ten years of experience with Webratio tool and WebML. The OMG adopted it as a standard in March 2013 [15].

IFML offers multiple advantages to the design of UIs for desktop, mobile, and web applications. One of the five artifacts the IFML specification provides is IFML visual syntax [4]. The latter, however, is simplified and well-known to developers, which discusses the development of various tools for creating IFML diagrams.

### C. Webratio Tool

Some technologies are designed for IFML model creation and code production for the creation of web and mobile applications. Webratio tool [16] focuses on rapid developments using model-driven development and code generation.

It is a commercial web development tool. When the Webratio tool was launched in 2001, it was built using WebML [17]. Later, the Webratio tool used IFML to replace WebML.

The entire process of modeling a web application is supported by the Webratio tool. Typically, one begins by developing the application's data model, or "domain model". This model defines all the data types that the web application will use.

Site views can be made once the domain model has been established. Each of these site views includes an IFML model. The developer can generate the designed web application after modeling it in IFML.

Last but not least, Webratio tool enables the direct deployment of the web application to a remote server.

As a tool, Webratio facilitates IFML web development from conceptual modeling through execution and also enables the entire web system generation with a click when the created IFML models are valid.

## III. RELATED WORK

We relate the previous works that discuss the application of the model-driven development approach to obtain graphical user interfaces.

The authors in [5] represent a methodology that permits a semi-automatic conversion from the CIM level to the PIM level using the MDA's approach, BPMN, and IFML standards. Several rules for semi-automating the conversion from the CIM level to the PIM level were developed. To achieve this an order management case study was provided to demonstrate the transformation strategy.

In [6] a technique for the model-driven creation of a Graphical User Interface for Internet Applications utilizing IFML was described. Model-driven engineering-related frameworks and technologies were used by the authors.

The construction of a prototype Qt/Taurus code generator based on the IFML standard and appropriate modeling tools, which are expanded to provide platform-specific code generation was described by the researchers in [7].

The goal is to enable low-code development in SKA GUI design, increasing the effectiveness, reliability, and coherency of the UI that is created.

A basic GUI use case is provided to fully illustrate the software development life cycle, starting with requirements and incorporating IFML modeling, Qt/Taurus automatic coding, interface evaluation, and validation.

The authors of this article [8] explored the modeling tools for IFML and presented a comparative analysis while taking into account several criteria. In this study, IFMLEdit.org, Webratio tool, IFML Editor, and MIA-studio were all studied and compared as potential IFML tools. Each tool has advantages and disadvantages.

This paper [9] suggested a modeling approach for creating user interfaces that are based on IFML. First, it describes the benefits of the model shifting process, the Webratio tool, and the extension of IFML to web applications.

Second, a detailed method for mapping IFML to the design environment is provided. The authors presented IFML for the web application, which can fully satisfy the expectations of the user. Its primary benefit is that it makes creating the project's front-end interface simple and quick.

AutoCRUD is a Webratio tool plug-in that aims the development of CRUD operations automatically by producing IFML specifications, the plug-in raises the efficiency of Web developers according to Rodriguez-Echeverria et al. [10].

In this work [11] authors provided a new method for creating the user interface for mobile applications, and implementing it with the Android operating system.

By establishing a language for the creation of graphical user interfaces, the Technology Neutral DSL (Domain-specific language) is designed to be cross-compiled to generate native code for a diversity of platforms.

The development of software applications incorporating a multi-experience User Interface is explored in this study through model-driven approaches [12].

The authors explained how elevating the abstraction level at which these interfaces are defined allows for quicker development, better deployment, and better integration of each interface with the rest of the software system and any other interfaces it may need to cooperate with.

They provide a new Domain Specific Language (DSL) for describing various CUI kinds and demonstrate how this DSL may be a component of an integrated modeling environment that can explain the relationships between the modeled CUIs and the other models of the system.

#### IV. METHODOLOGY

The CIM model is intact by the application's technical details and includes both business process requirements and client needs.

BPMN is a standard for creating business processes. Application interface modeling is done using IFML, it comes

with a suite of tools for modeling user interaction and computer program behavior graphically.

In our methodology, we used the Eclipse tool to transform the BPMN model (CIM level) into the IFML model (PIM level) according to the MDA approach.

First of all, we created the BPMN and IFML metamodels shown in Fig. 2, Fig. 3, and Fig. 4 in Eclipse tool, then the creation of the BPMN model from the BPMN business process as a ".xmi" file from the BPMN metamodel, and we finally obtain the IFML model in ".xmi" file too by applying the transformation rules shown in Fig. 1 written in ATL language [5].

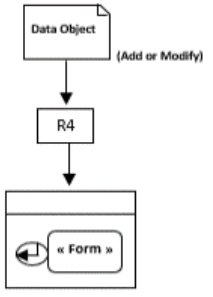
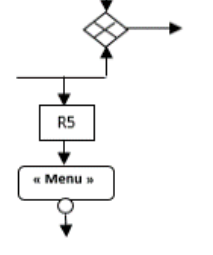
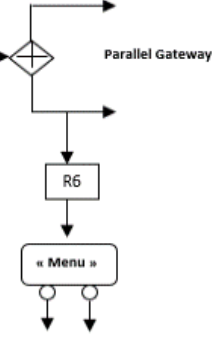
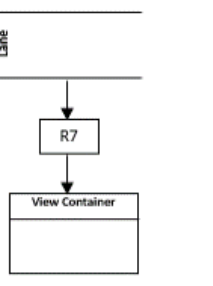
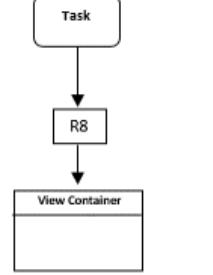
Lastly, we import the xmi file obtained to the commercial tool Webratio to generate the user interfaces automatically at the PSM level.

#### A. The CIM Level to the Pim Level Conversion Rules

BPMN models to IFML models transformation rules are shown in Fig. 1 every converting rule is defined in ATL and human language and represented by a schema.

Firstly, it is necessary to define the PIM level metamodel and the CIM level metamodel to describe ATL's transformation rules.

Human Language	Graphical representation	Transformation rule in ATL
-Each "State" of "Data Object" becomes "View Component"		<pre> Rule R 1{   from   stt : MMbpmn!StateObject     (stt.isTransformableStateObject ())   to   vcp : MMifml!ViewComponent{     Name &lt;- stt.name   } } </pre>
-Each "Display" state of "Data Object" becomes "View Container" with "List" view component plus "On Select Event"		<pre> Rule R 2{   from   stt : MMbpmn!StateObject     (stt.isTransformableStateObject () and   stt.isDisplay())   to   cn : MMifml!Container {     Name &lt;- '«WebPage»'+stt.name)   vcp : MMifml!ViewComponent(name&lt;-     '«List»',   ContainerBelongs&lt;-cn.name)   Osev : MMifml!OnSelectEvent{     ViewComponentBelongs&lt;-vcp.name)   } } </pre>
-Each "Delete" state of "Data Object" becomes " View Container" with "Delete" action plus simple event		<pre> Rule R 3{   from   stt : MMbpmn!StateObject     (stt.isTransformableStateObject () and   stt.isDelete())   to   cn : MMifml!Container {     Name &lt;- '«WebPage»'+stt.name)   ac : MMifml!Action (name&lt;- 'Delete',   ContainerBelongs&lt;-cn.name)   sev : MMifml!SimpleEvent(     ViewComponentBelongs&lt;-ac.name)   } } </pre>

<p>-Each "Add or Modify" state of "Data Object" becomes "View Container" with "Form" view component plus "On submit event"</p>		<pre> Rule R 4{   from   stt : MMBpmn!StateObject   (stt.isTransformableStateObject () and   (stt.isAdd() or stt.isModify()))   to   cn : MMifm!Container (   name &lt;- '&lt;WebPage&gt;'+stt.name)   vcp : MMifm!ViewComponent(   name&lt;-'Form'+ stt.name,   containerBelongs&lt;-cn.name)   osev : MMIFML!OnSubmitEvent(   ViewComponentBelongs&lt;-ac.name)   } </pre>
<p>-Each "Exclusive Gateway" becomes "Menu" view component with simple event corresponding transition in output "Exclusive Gateway"</p>		<pre> Rule R 5{   from   exl : MMBpmn!Exclusive   (exl.isTransformableEclusive ())   to   vcp: MMIFML!ViewComponent (   name&lt;- '&lt; Menu &gt;'+exl.name })   Rule R {   from   fl : MMBPMN!Flow   (fl.isTransformableFlow() and   fl.OutputisDecisionState())   To   MMIFML!SimpleEvent(name&lt;-fl.name,   ViewComponentBelongs&lt;-   Fl.NameDecisionStateOutput() )   } </pre>
<p>-Each "Parallel Gateway" becomes "Menu" view component with "Events" corresponding transition in output "Parallel Gateway"</p>		<pre> Rule R 6 {   from   prl : MMBpmn!Parallel   (prl.isTransformableParallel ())   to   vcp: MMIFML!ViewComponent (   name&lt;- '&lt; Menu &gt;'+prl.name })   Rule R {   from   fl : MMBPMN!Flow   (fl.isTransformableFlow() and   fl.OutputisDecisionState())   To   MMIFML!SimpleEvent(name&lt;-fl.name,   ViewComponentBelongs&lt;-   Fl.NameDecisionStateOutput() )   } </pre>
<p>-Each "Lane" becomes "View Container"</p>		<pre> Rule R 7{   from   ln : MMBpmn!Lane   (ln.isTransformableLane())   to   vcn : MMifm!ViewContainer (   Name &lt;- ln.name   )   } </pre>
<p>-Each "Task" becomes "View Component"</p>		<pre> Rule R 8{   from   tsk : MMBpmn!Task   ((not tsk.isManual()) and   (tsk.isTransformableTask()))   to   vcn : MMifm!ViewContainer (   Name &lt;- tsk.name   )   } </pre>

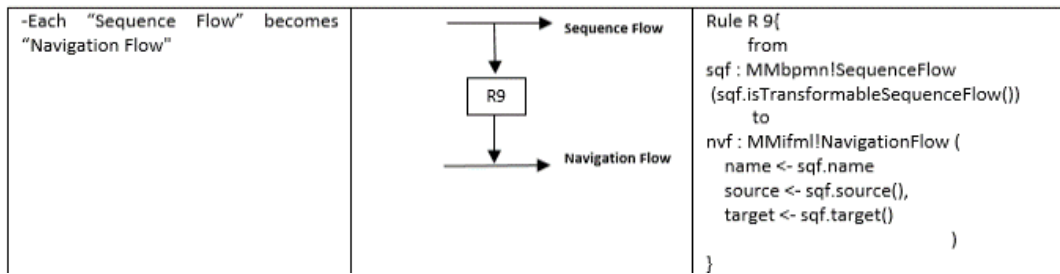


Fig. 1. CIM level to the PIM level transformation rules.

**B. CIM Level: BPMN Metamodel**

At the CIM level, BPMN was utilized to describe the business process because business processes are computationally independent, the OMG standard for business process modeling is named BPMN. Fig. 2 displays the BPMN metamodel [18].

**C. PIM Level: IFML Metamodel**

IFML Metamodel that uses the UML metamodel's main data types provides a set of UML metaclasses as the foundation for IFML metaclasses and presumes that the IFML ContentModel is defined in UML [19], [20].

The IFML metamodel is illustrated in Fig. 3 and Fig. 4.

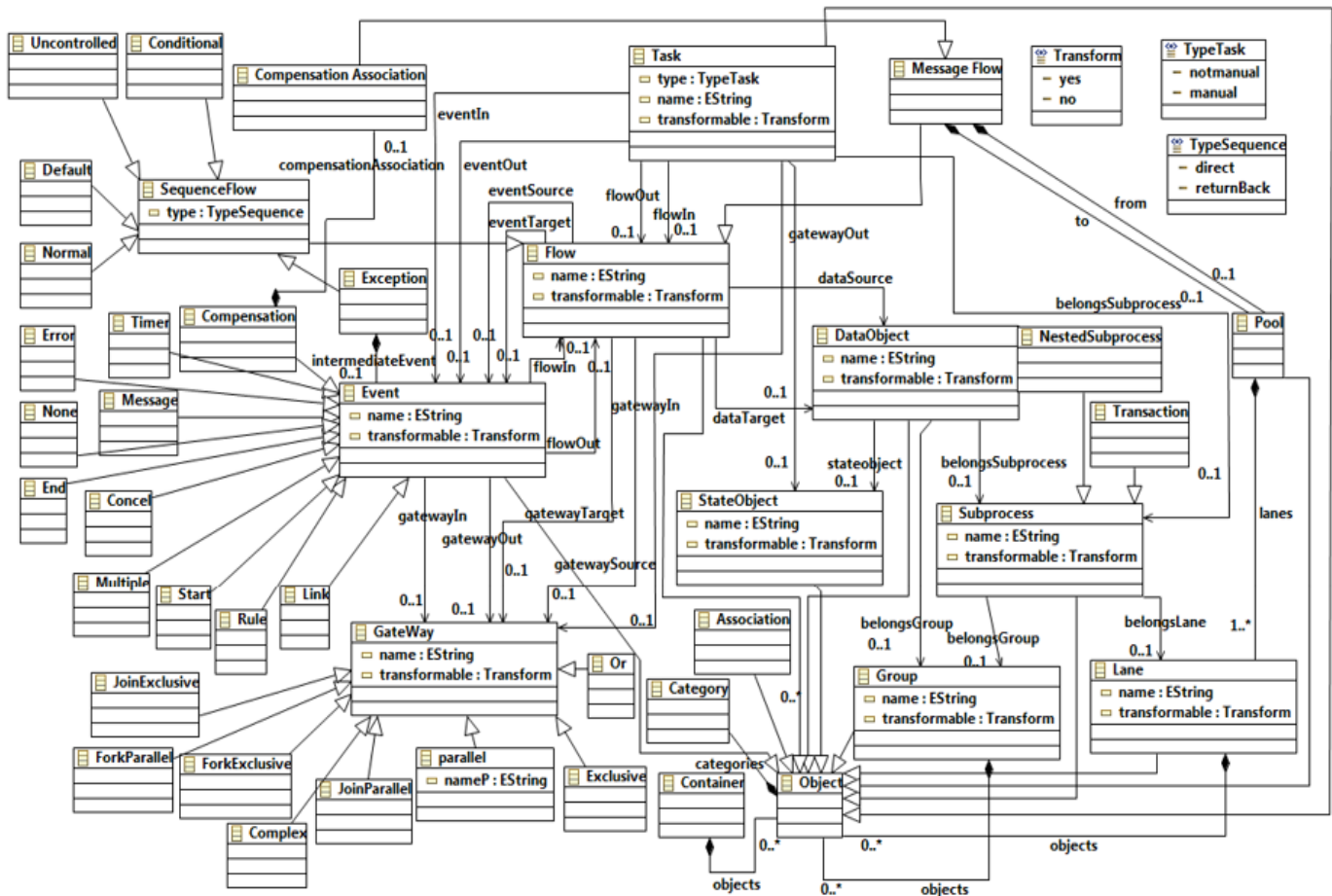


Fig. 2. The BPMN metamodel.

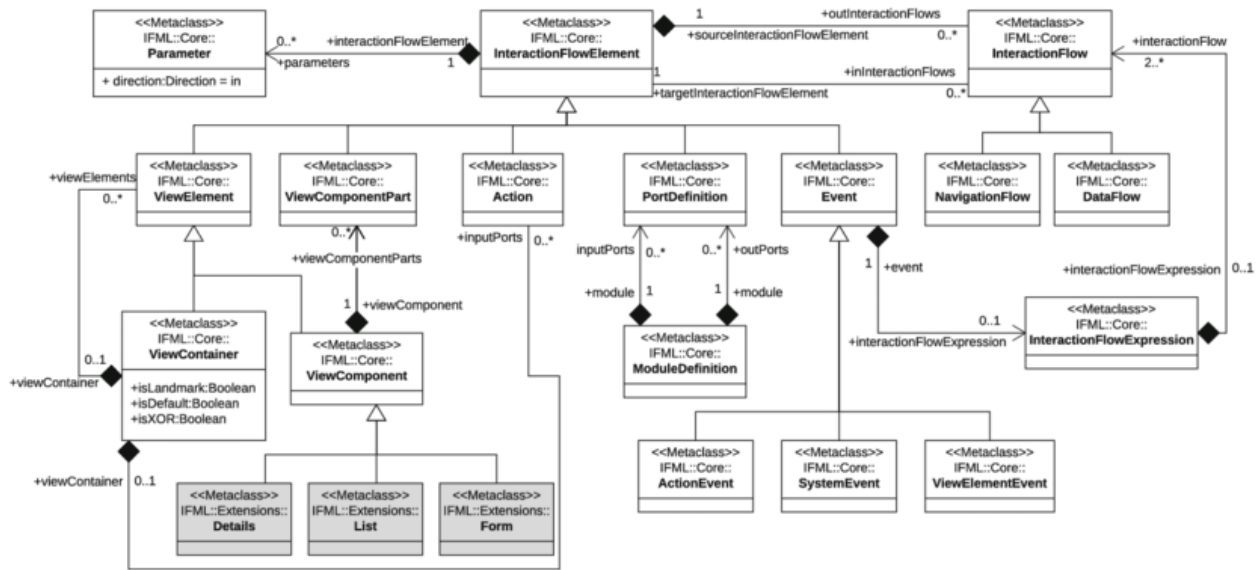


Fig. 3. IFML metamodel 1.

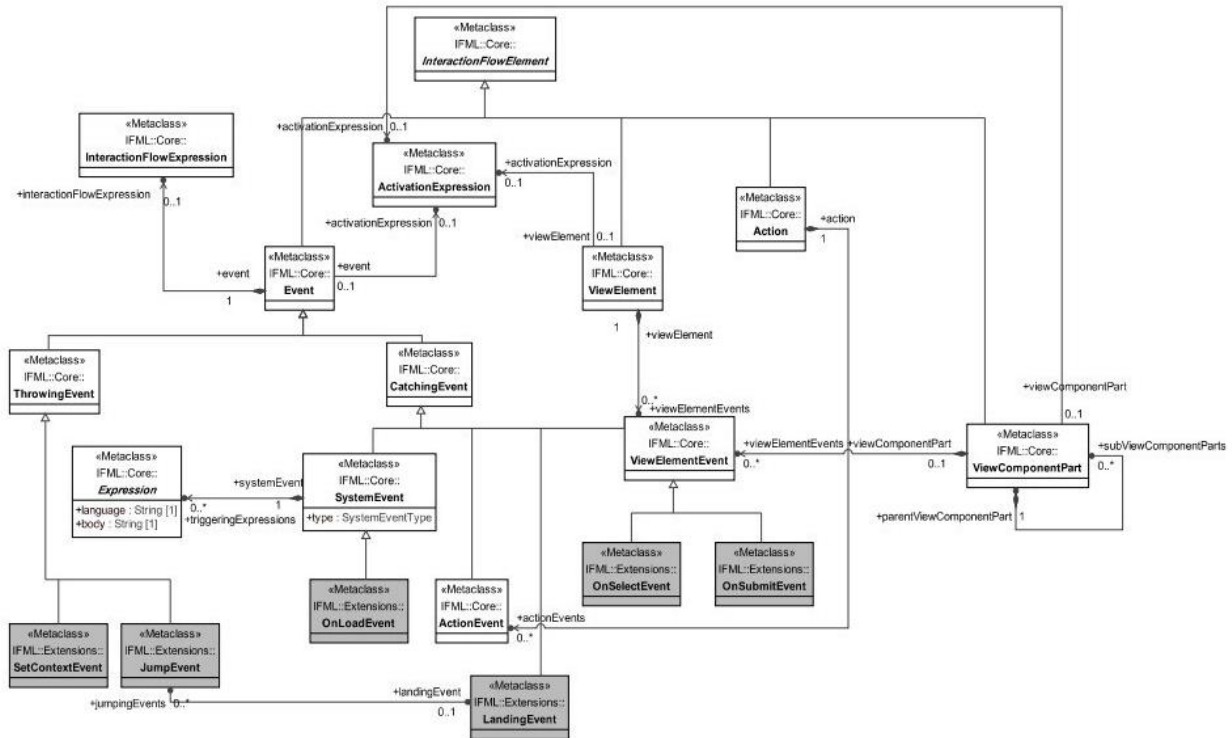


Fig. 4. IFML metamodel 2.

## V. CASE STUDY

The after-sales service is described, in which a customer can make CRUD operations (Create, Read, Update, and Delete) relating to a complaint if there is a problem regarding his order.

The process begins with authentication using a username and password. If the client is already registered on the

platform, he has access to a menu with four features, including add, modify, delete, and consult a complaint. If not, he must register to use the remaining features.

### A. CIM Level

The CIM level is illustrated by the BPMN diagram and BPMN business process, the first one is represented in Fig. 5, and the second is displayed in Fig. 6.

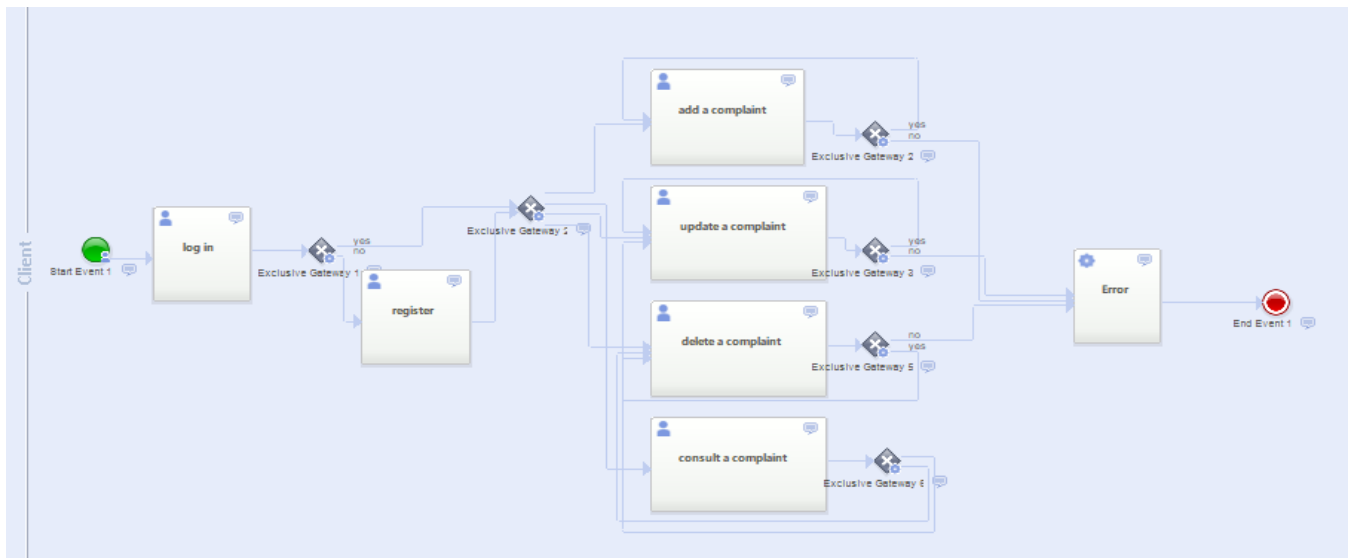


Fig. 5. BPMN diagram of the after-sales service.

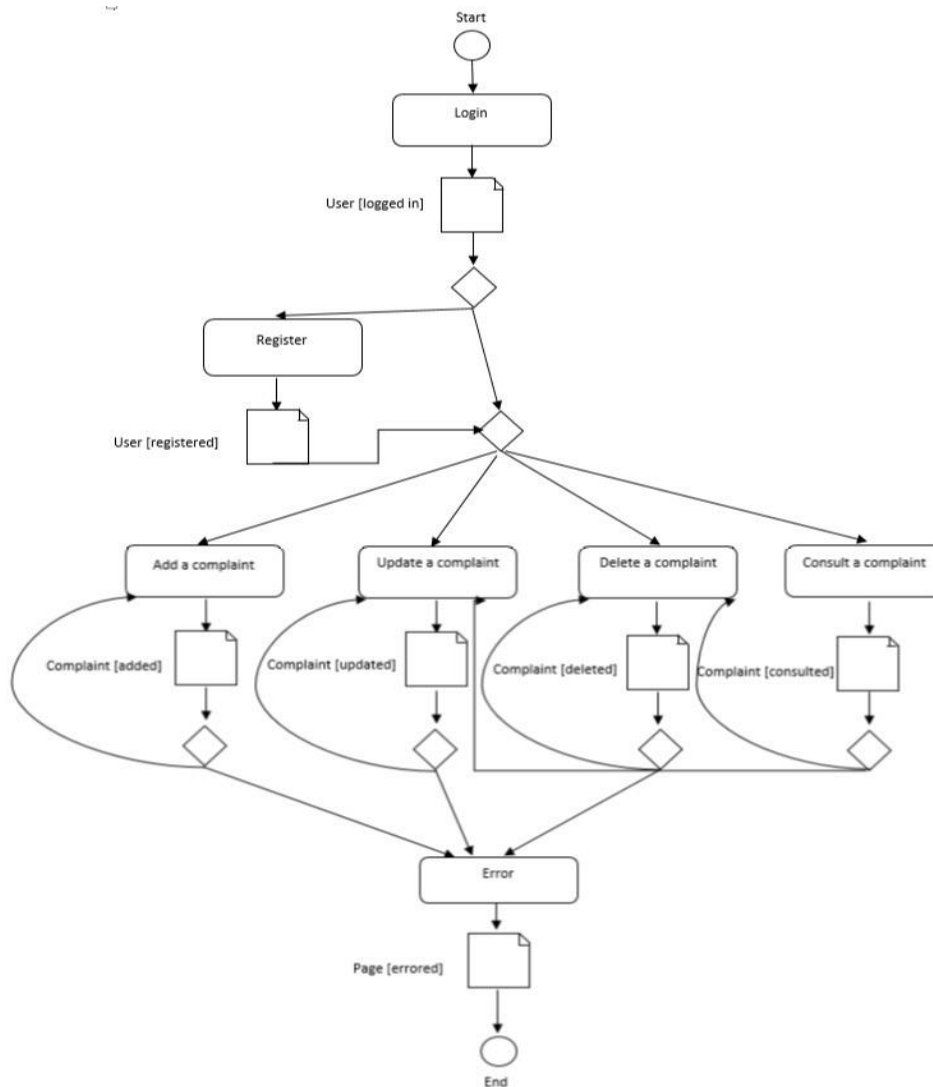


Fig. 6. BPMN business process diagram of the after-sales service.



### B. PIM Level

Webratio is the most powerful commercial tool available among others [8]. Some of the characteristics that motivated us to choose to work with this tool are:

- Provides graphical modeling; the IFML components are expanded further by this tool, so that users can design IFML diagrams for web applications.
- Based on the Eclipse and Webratio tools respond quickly.
- It is suitable for academic researchers.

We obtained the after-sales service front-end interfaces designed with the IFML language, by applying the shift rules from CIM to PIM. [5].

IFML development process involves modeling the application's domain first, then creating an IFML model using IFML language, and then converting the created model into a software system.

1) *Domain model*: A clear and accessible overview of information related to the application domain is provided by the domain model.

We don't suggest a novel modeling language for domain model modeling because a UML-based OMG standard is IFML; rather, we utilize UML class diagrams.

The case study's Domain model is described in Fig. 7 we added a table named complaint, User, Group, and Module are tables provided by default by the Webratio tool.

2) *IFML model*: In our case study, we include only two important elements in the example: the login/registration function and the client's space.

The public is the application's main page, allowing the user to authenticate himself to access the proposed features (Login page) or to register if it is his first connection to the application.

Otherwise, an error message is displayed, Fig. 8 shows the "Public" site view.

The site view Public contains an action definition (a description of a business logic that could be triggered by an event in an IFML model.) [21] "Register", which will allow the creation of a new user in the database and Fig. 9 shows its modeling.

The action definition Register begins with the input port, which receives the parameters from the user's registration form. The parameters are passed to the "Create User" operation which is based on the Entity "User."

We utilized a Data Flow to bind the information from the Input Port to the Output Port, the information between the input port and the corresponding "Create User", and the latter to the output port.

When these operations are completed successfully, the Action Definition must exit via the "Success" OK Port, otherwise, the "KO Port" handles the failed execution.

The complaint is the space dedicated to the user where he can perform one of the operations provided by CRUD (Create, Read, Update, or Delete a complaint).

In Fig. 10, the interface modeling is displayed.

In the site view Complaint, four areas were used to model CRUD operations. Each area is composed of pages and actions definitions, and also utilized view components (List, Form).

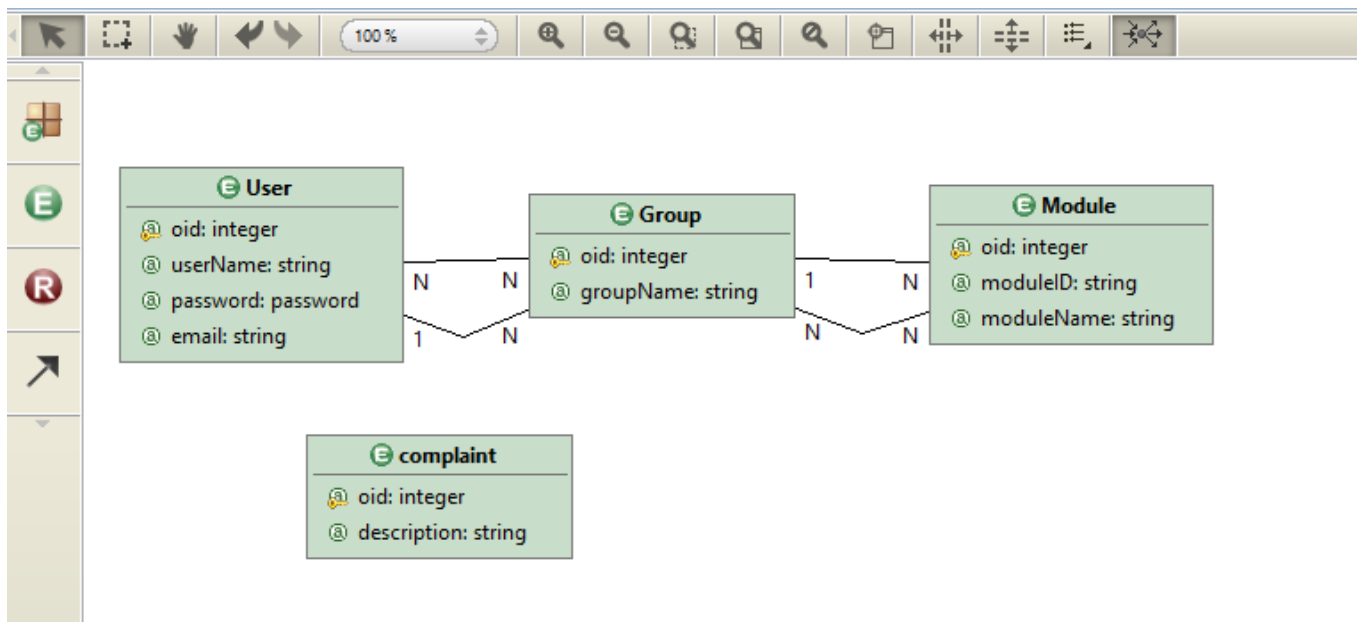


Fig. 7. Domain model of the after-sales service.



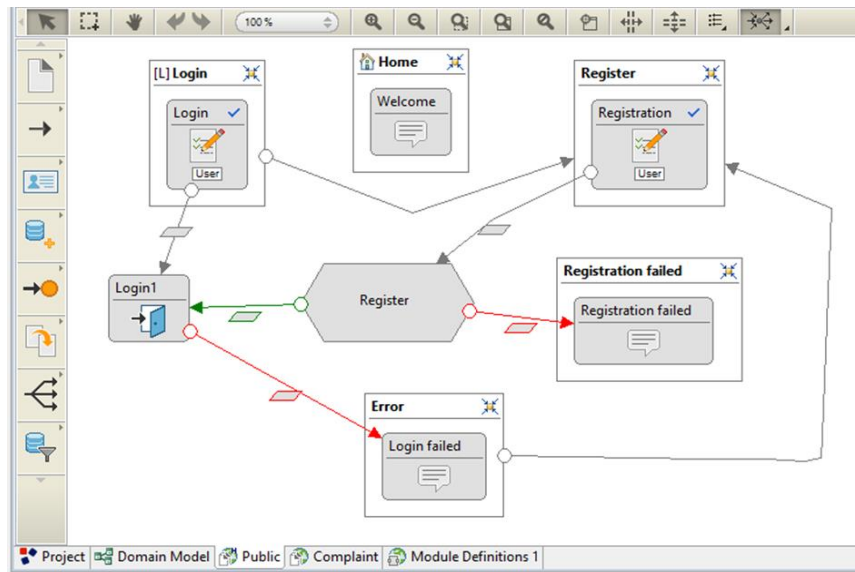


Fig. 8. Site view "Public".

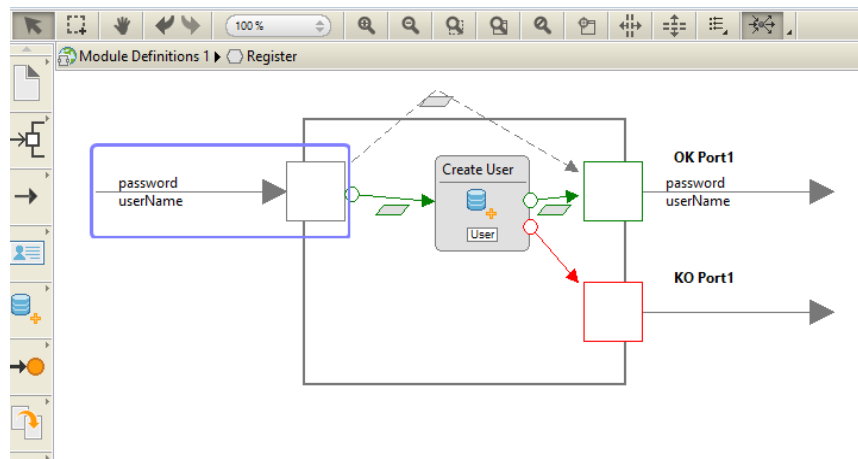


Fig. 9. Action definition "Register".

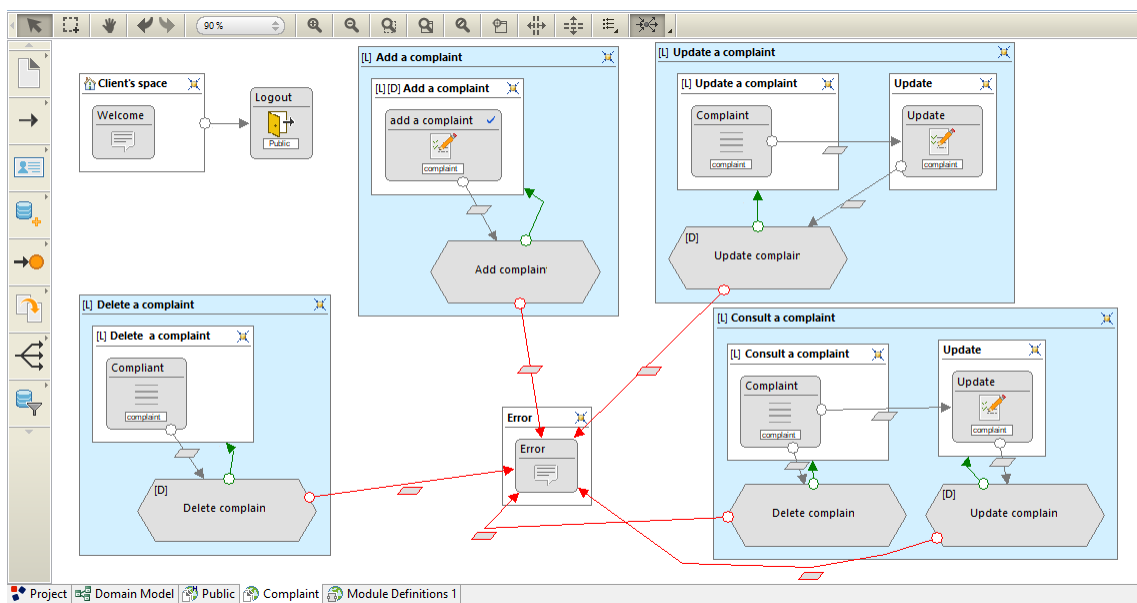


Fig. 10. Site view "Complaint".

To make the connection between the different pages and actions definitions we used flows (flow, OK flow, and KO flow) with parameters binding between source and target elements.

For example, to add a complaint we have a page named complaint containing a form. When the user presses the button "save", the action definition Add a complaint is triggered, using a flow going from the form to the action definition, by editing parameters binding.

If the addition is done successfully we return to the Add a complaint page, otherwise we are redirected to an error page.

### C. PSM Level

In this paragraph, we will present some of the graphical user interfaces obtained by generating the application of the IFML model using the Webratio tool.

Fig. 11 shows the authentication page which provides the user to authenticate himself through a username and password.

Otherwise, he must register first by entering his personal information, and Fig. 12 displays this operation.

Another interface where the user can add a complaint is shown in Fig. 13.

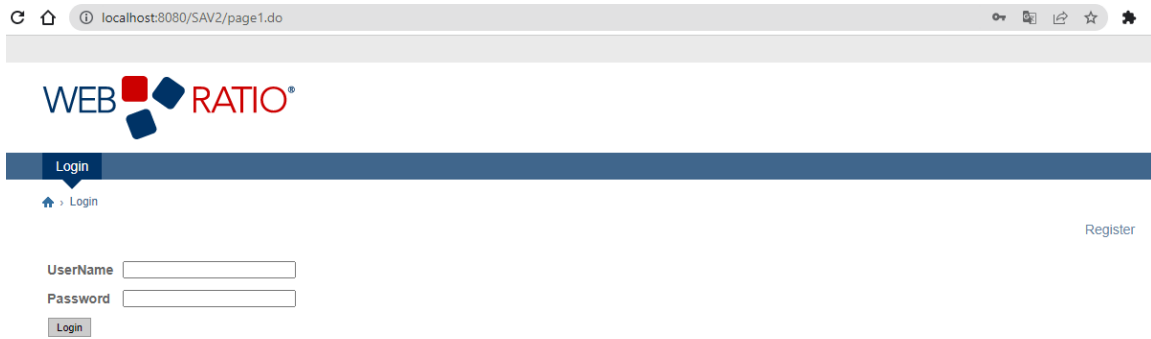


Fig. 11. "Login" graphical interface.



Fig. 12. "Register" graphical interface.

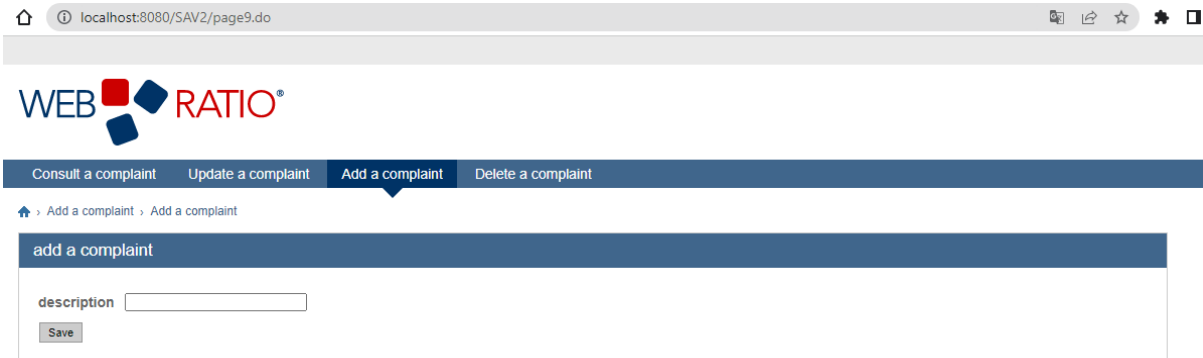


Fig. 13. "Add a complaint" graphical interface.

## VI. DISCUSSION AND LIMITATIONS

We may infer from an analysis of the works shown in Table I, that they can be divided into two groups. Those who were interested in the CIM to PIM shift and those who were interested in the PIM to PSM shift.

We can say that the majority of works use UML diagrams to present the different levels of the MDA approach, except [25] which used the BPMN standard that is utilized for business process modeling.

The UML class diagram is often utilized in most PIM to PSM transformations, IFML as a recommended standard for the modeling front-en interfaces at the PIM level is not widely used in the studied papers apart in [30].

The work [31] treated the model conversions in the MDA approach from the CIM level to the PSM via the PIM level, by using two models the first is for requirement and the second is a mathematical model represented by a graph.

The major advantage of our work which differs from other researches [33-40] is the fact that it covers all the model

transformations process in the MDA's approach. Starting from the CIM via the PIM level up to the PSM level, utilizing two standards BPMN and IFML, and a powerful commercial tool Webratio.

In our case study, we have limited it to CRUD operations, we can still extend it to cover other aspects of after-sales service, also the process of our methodology is semi-automatic in the importation of the xmi file in Webratio tool.

The CIM level to the PIM level shift is implemented in the Eclipse tool automatically, the IFML model is obtained in an xmi format [32] and imported into the Webratio tool to finally get the user interfaces.

The advantage of using IFML modeling is that the front-end interface of the program can be created quickly also the application is generated with a click.

But the technology has some disadvantages, for example, even though IFML modeling is still relatively a new technology, it is still best suited for modeling small programs rather than programs for large and medium-sized businesses.

TABLE I. COMPARISON OF PREVIOUS WORKS

Studied Papers	CIM to PIM shift		PIM to PSM shift	
	Reached	Illustration	Reached	Illustration
Rhazali et al. [22]	Yes	UML diagrams		
Kharmoum et al. [23]	Yes	UML class diagram		
Arrhioui et al. [24]	Yes	UML class diagram		
Khlif et al. [25]	Yes	BPMN		
Erraissi et al. [26]			Yes	UML class diagram
Deeba et al. [27]			Yes	UML class diagram
Srai et al. [28]			Yes	UML class diagram
Betari et al. [29]			Yes	UML class diagram
Koren et al. [30]			Yes	IFML
Zhang et al. [31]	Yes	Business-oriented Conceptual Model	Yes	Graph
Our methodology	Yes	BPMN	Yes	IFML

## VII. CONCLUSION

Developing software is easier with MDA's approach because it generates applications automatically from models in a shorter period and with absolutely no source code involvement.

Typically, web applications are built using Webratio's tool robust architecture, it is a consolidated industrial reality that enables any model to be created. In this sense, we suggested this work which highlighted the different model transformations.

Our methodology adheres to the MDA principles and suggests defining the various MDA levels' characteristics, as well as the CIM level to the PIM level shift rules utilizing two standards BPMN and IFML, till automatically generating the user interfaces utilizing the Webratio tool.

This is done by creating the two metamodels BPMN and IFML in Eclipse tool, after we elaborated the BPMN model of the case study then, by applying the shift rules in ATL language, we obtain the IFML model under .xmi format.

Finally, we imported this file in Webratio tool to get the graphical user interfaces.

The objective of improvement is the creation of a tool that will allow the model transformations from the CIM level until the obtention of the graphical user interfaces that will automatize the whole process of model transformations.

## REFERENCES

- [1] OMG, MDA Guide Version 2.0., 2014, [Online]. Available at: <http://www.omg.org/mda>.
- [2] M. Brambilla, A. Mauri and E. Umuhoza, "Extending the Interaction Flow Modeling Language (IFML) for Model Driven Development of Mobile Applications Front End", Awan, I., Younas, M., Franch, X., Quer, C. (eds) Mobile Web Information Systems. MobiWIS Lecture Notes in Computer Science, 2014, pp 176-191, vol 8640. Springer, Cham. DOI: 10.1007/978-3-319-10359-4\_15.
- [3] M. Brambilla, P. Fraternali. Interaction Flow Modeling Language: Model-Driven UI Engineering of Web and Mobile Apps with IFML (1st. ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2014, ISBN: 9780128001080, ISBN: 9780128005323. DOI:10.5555/2785641.
- [4] R. Acerbis, A. Bongio, M. Brambilla and S. Butti, Model-Driven Development Based on OMG's IFML with WebRatio Web and Mobile Platform. In: Cimiano, P., Frasinca, F., Houben, GJ., Schwabe, D. (eds)

- Engineering the Web in the Big Data Era. ICWE 2015. Lecture Notes in Computer Science(), 2015, vol 9114. Springer, Cham. DOI: 10.1007/978-3-319-19890-3\_39.
- [5] A. Sajji, Y. Rhazali and Y. Hadi, "A methodology for transforming BPMN to IFML into MDA", *Bulletin of Electrical Engineering and Informatics*, 11: 2773-2782, number 5,2022, pp. 2773-2782. DOI:10.11591/eei.v11i5.397.
- [6] S. Roubi , M. Erramdani and S. Mbarki, "A Model Driven Approach for generating Graphical User Interface for MVC Rich Internet Application", *Computer and Information Science*. 9. 10.5539/cis.v9n2p91. Press, Japan,2016, pp: 249-256. DOI: 10.1109/PNPM.1989.68558.
- [7] M. Brambilla , R. Cirami, M. Gasparini, A. Marassi and S. Pavanetto, "Code Generation based on IFML for the User Interfaces of the Square Kilometre Array (SKA)", *Proc. ICALEPCS'19*, New York, NY, USA, 2019, pp. 1307-1311. DOI: 10.18429/JACoW-ICALEPCS2019 WEPHA093.
- [8] N. Laaz, K. Wakil, S. Mbarki and D.N. Jawawi , "Comparative Analysis of Interaction Flow Modeling Language Tools". *Journal of Computer Science*, 14(9), 1267-1278, 2018. DOI: 10.3844/jcssp.2018.1267.1278.
- [9] K. Rong, X. Liu, " IFML-Based Web Application Modeling", *Procedia Computer Science*, 166, 2020 pp: 129- 133,ISSN: 1877-0509. DOI: 10.1016/j.procs.2020.02.034.
- [10] R. Rodriguez-Echeverria, C. Preciado, J. Sierra, M. Conejero , F. Sanchez-Figueroa,"AutoCRUD: Automatic generation of CRUD specifications in interaction flow modelling language", *Science of Computer Programming*,168, 2018, pp 165-168, ISSN 0167-6423 DOI: 10.1016/j.scico.2018.09.004.
- [11] M. Lachgar, A. Abdali, " Generating Android graphical user interfaces using an MDA approach", *Third IEEE International Colloquium in Information Science and Technology (CIST)*, 2014, pp. 80-85, DOI: 10.1109/CIST.2014.7016598.
- [12] E. Planas, G. Daniel, M. Brambilla, "Towards a model-driven approach for multi-experience AI-based user interfaces", *Softw Syst Model* 20, pp. 997–1009, 2021. DOI: 10.1007/s10270-021-00904-y.
- [13] M. Dumas, M. La Rosa, J. Mendling, A. Hajo, H. Reijers, *Fundamentals of Business Process Management*, London, 2018, ISBN: 978-3-662-56508-7. DOI 10.1007/978-3-642-33143-5.
- [14] OMG, *Business Process Model, and Notation (BPMN)*, 2013, [Online]. Available at: <https://www.omg.org/spec/BPMN/2.0/PDF>.
- [15] C. Bernaschina, M. Brambilla, A. Mauri, E. Umuhoza , "A Big Data Analysis Framework for Model-Based Web User Behavior Analytics", Cabot, J., De Virgilio, R., Torlone, R. (eds) *Web Engineering*, Lecture Notes in Computer Science(), ICWE ,2017, vol 10360. Springer, Cham. DOI: 0.1007/978-3-319-60131-16.
- [16] R. Acerbis et al., "WebRatio, an Innovative Technology for Web Application Development" ,Koch, N., Fraternali, P., Wirsing, M. (eds) *Web Engineering. ICWE. Lecture Notes in Computer Science*, vol 3140. Springer, Berlin, Heidelberg, 2004. DOI: 10.1007/978-3-540-27834-4\_90.
- [17] S. Ceri, P. Fraternali, A. Bongiotto, "Web Modeling Language (WebML): a modeling language for designing Web sites", *Computer Networks*, 33, 2000, pp. 137-157,ISSN :1389-1286. DOI: 10.1016/S1389-1286(00)00040-2.
- [18] Y. Rhazali, Y. Hadi, A. Mouloudi, "A Methodology of Model Transformation in MDA: from CIM to PIM", *International Review on Computers and Software (I.RE.CO.S.)*, vol. 10, no. 12, 2015, pp. 1186-1201. DOI: 10.15866/irecos.v10i12.8088.
- [19] M. Brambilla, E. Umuhoza, R. Acerbis, "Model-driven development of user interfaces for IoT systems via domain-specific components and patterns", *Internet Serv Appl* 8, 14,2017. DOI: 10.1186/s13174-017-0064-1.
- [20] OMG, *Interaction Flow Modeling Language*, 2015, [Online]. Available at: <https://www.omg.org/spec/IFML/1.0/Beta1/PDF>.
- [21] Acerbis, S., *Webratio* , 2001, [Online]. Available at: <https://my.webratio.com/learn/content?nav=65&link=oln208a.redirect&so=pcu1a>.
- [22] Y. Rhazali, Y. Hadi, I. Idriss Chana, M. Lahmer, A. Abdallah Rhattoy, "A Model Transformation in Model Driven Architecture from Business Model to Web Model", *IAENG International Journal of Computer Science*, 45:1, *IJCS\_45\_1\_16*, Vol. 45 Issue 1, pp104-117, vol. 45, no.1, 2018 [https://www.iaeng.org/IJCS/issues\\_v45/issue\\_1/IJCS\\_45\\_1\\_16.pdf](https://www.iaeng.org/IJCS/issues_v45/issue_1/IJCS_45_1_16.pdf).
- [23] N. Kharmoum, S. Ziti, Y. Rhazali, F. Omary, "An Automatic Transformation Method from the E3value Model to IFML Model: An MDA Approach", *Journal of Computer Science*, 15(6), 800-813, 2019. DOI: 10.3844/jcssp.2019.800.813.
- [24] K. Arrhioui, S. Mbarki , M. Erramdani, "Applying CIM-to-PIM Model Transformation for Development of Emotional Intelligence Tests Platform", *International Journal of Online and Biomedical Engineering (iJOE)*, 14(08), pp. 160–168,2018. DOI: 10.3991/ijoe.v14i08.8747.
- [25] W. Khlif , N.E. Ayed , H. Ben-Abdallah, "From a BPMN Model to an Aligned UML Analysis Model". *ICSOFT*, Porto, Portugal, 2018, pp. 623-631,. DOI: 10.5220/0006866606570665.
- [26] A. Erraissi, M. Banane, A. Belangour, M. Azzouazi," Big Data Storage using Model Driven Engineering: From Big Data Meta-model to Cloudera PSM meta-model", *International Conference on Data Analytics for Business and Industry: Way towards a Sustainable Economy (ICDABI)*, pp. 1-5,2020. DOI: 10.1109/ICDABI51230.2020.9325674.
- [27] F. Deeba, S. Kun, M. Shaikh, F. A. Dharejo, S. Hayat ,P. Suwansrikham, "Data transformation of UML diagram by using model driven architecture" ,*IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*,2018,pp. 300-303. DOI: 10.1109/ICCCBDA.2018.8386531.
- [28] A. Srari, A. F. Guerouate, N. Berbiche, H. HilalDrissi, "Generated PSM Web Model for E-learning Platform Respecting n-tiers Architecture", *International Journal of Emerging Technologies in Learning (IJET)*, 12(10), pp. 212–220, 2017. DOI: 10.3991/ijet.v12i10.7179.
- [29] O. Betari, M. Erramdani, S. Roubi, K. Arrhioui ,S. Mbarki, "Model Transformations in the MOF Meta-Modeling Architecture: From UML to CodeIgniter PHP Framework", Rocha, A. Serhini, M., Felgueiras, C. (eds) *Europe and MENA Cooperation Advances in Information and Communication Technologies. Advances in Intelligent Systems and Computing*, vol 520. Springer, Cham, 2017, pp. 227–234. DOI: 10.1007/978-3-319-46568-5\_23.
- [30] I. Koren, R. Klamma, "The Exploitation of OpenAPI Documentation for the Generation of Web Frontends", In *Companion Proceedings of the Web Crence 2018 (WWW '18)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 2018, pp. 781–787. DOI: 10.1145/3184558.3188740.
- [31] L. Zhang, H. Pingaud, E. Lamine, F. Fontanili, C. Bortolaso , M. Derras, "Model-Driven Approach to Transform Business Vision-Oriented Decision-Making Requirement into Solution-Oriented Optimization Model", Fujita, H., Fournier-Viger, P., Ali, M., Wang, Y. (eds) *Advances and Trends in Artificial Intelligence. Theory and Practices in Artificial Intelligence. IEA/AIE. Lecture Notes in Computer Science ()*, vol 13343. Springer, Cham, 2022, pp. 211–225. DOI: 10.1007/978-3-031-08530-7\_18.
- [32] OMG, "XML Metadata Interchange (XMI) Specification." 2015, [Online]. Available at: <https://www.omg.org/spec/XMI/2.5.1/PDF>.
- [33] A. Sajji, Y. Rhazali, Y. Hadi, "An Approach to Automate Generation of Graphical User Interfaces Through IFML", *Procedia Computer Science*, Vol 201, pp. 621-626, 2022, ISSN 1877-0509, DOI : 10.1016/j.procs.2022.03.081.
- [34] M. Ayadi, Y. Rhazali, M. Lahmer , "A Proposed Methodology to Automate the software manufacturing through Artificial Intelligence (AI)", *Procedia Computer Science*, Vol 201, pp. 627-631, 2022, ISSN 1877-0509 DOI: 10.1016/j.procs.2022.03.082.
- [35] S. Nasiri, Y.Rhazali, M. Lahmer, A. Adadi, "From User Stories to UML Diagrams Driven by Ontological and Production Model". *International Journal of Advanced Computer Science and Applications*, 12 (6), pp. 333 – 340, 2021 DOI: 10.14569/IJACSA.
- [36] N. Kharmoum, S. Ziti, Y. Rhazali,F. Omary, "A method of model transformation in mda approach from e3value model to bpmn2 diagrams in cim level". *IAENG International Journal of Computer Science*, vol. 46, no. 4, pp. 1–17, 2019. DOI: 10.47839/ijc.18.3.1524.
- [37] S. Nasiri, Y. Rhazali, A. Adadi, M. Lahmer, "Generation of User Interfaces and Code from User Stories". In: Joshi, A., Mahmud, M., Ragel, R.G. (eds), presented at *Information and Communication Technology for Competitive Strategies (ICTC 2021)*. Lecture Notes in

- Networks and Systems, vol 400, pp. 397-409, Springer, Singapore, ISBN 978-981-19-0094-5, 2023 DOI: 10.1007/978-981-19-0095-2\_38.
- [38] Y. Rhazali, Y. Hadi, A. Mouloudi, "Model Transformation with ATL into MDA from CIM to PIM Structured through MVC", *Procedia Computer Science*, 83, pp. 1096-1101, 2016. DOI:10.1016/j.procs.2016.04.229.
- [39] N. Kharmoum, S. Ziti, Y.Rhazali, F. Omary, "An automatic transformation method from the e3value model to uml2 sequence diagrams: An MDA approach". *International Journal of Computing*, vol. 18, no. 3, pp. 316–330, 2019 DOI:10.47839/ijc.18.3.1524.
- [40] Y. Rhazali, Y. Hadi, A. Mouloudi, (2016b), "CIM to PIM Transformation in MDA: from Service- Oriented Business Models to Web-Based Design Models", *International Journal of Software Engineering and Its Applications*. Vol. 10, no. 4, pp.125-142. DOI: 10.14257/ijseia.2016.10.4.13.