Unified Modeling Language extensions for modeling user-oriented, multi-channel access CRM systems

Grzegorz Futa[*]

Department of Applied Computer Science, Institute of Computer Science, M. Curie Sklodowska University, 20-031 Lublin, Pl. M. Curie Sklodowskiej 1, Poland

Abstract

In the advances of Internet technologies in recent years, Electronic Commerce CRM systems have gained much attention as a major theme for companies to maintain their competitiveness. The research shows that the effective customer relationship management is the major source for customer retention and gaining over new ones. On the other hand, modern technology allows to receive information through different channels (Internet, phone, WAP). Therefore analysts are forced to use faster, more reliable methods for system modeling. The author proposes a new method for modeling Customer Relationship Management systems. The UML new extensions are introduced. The customer-oriented and multichannel access patterns aim at improvement of system modeling with the high level of abstraction. This paper identifies and analyses the main advantages of language additions and compare them to object-oriented modeling with the pure UML patterns.

1. Introduction

There are many publications related to the CRM information systems that discuss personalization methods (see e.g. [1-4]) and customer decision support techniques [5,6], so there are many technical solutions. Nevertheless, analysis and design methods for the CRM information systems may significantly affect system efficiency and system requirements satisfaction. It is especially important in the situation where a company customer can access required information using different multiple channels. Lin and Lee [7] proposed the object-oriented modeling method for CRM IS that prospects customers under their preferable execution environments. That can be treated as a step towards development of customer oriented, multi-channel modeling methodologies.

In general, the system analysis methods can be divided into three kinds of approaches: function-, data- and object-oriented methodologies [8] where the object-oriented ones are especially important because of their advantages of object-oriented techniques (inheritance, abstracting, reuse, etc.). Nevertheless,

---

[*]*E-mail address*: grzegorz.futa@umcs.lublin.pl

CRM IS concerns their effectiveness in their maintainability for satisfying complex, parametrical and changeable customers' requirements. It often happens that requirements and needs depend on the way the user accesses the information system. Furthermore, the possibility of usage of certain functionality requires analysis or updating the data possessed by other systems. The UML [9,10] and methodologies based on it [8] offer use-case driven methods ascertained by researchers as the robust process. Huge possibilities and multi-purpose symbols make the analysis process difficult for unaccustomed users. In some cases, there is a need for describing the system on high level of abstraction where some technical aspects are hidden.

The purpose of this work is to introduce a number of UML extensions that simplify the analysis and modeling of user-oriented, multi-channel access, integrated information systems.

This paper shortly introduces the proposed extensions. The sections below are organized as follows. Definitions explain the concepts used in this paper. Next three subsections present and describe the proposed symbols, their meaning and usage possibilities. The last section includes the discussion and conclusions.

2. UML extensions

It often happens that during the system modeling, analysts have to introduce their work to the customers as well as to the system designer. In most cases, customers are not familiar with UML notation. The high level of abstraction and unknown symbols can lead to misunderstanding and missing the point of requirement analysis. Secondly, system designers are forced to model the software appropriate for given application.

The UML language [10] offers general purpose modeling notations. In some cases, they can be hard to understand by the customers. On the other hand, the usage of the stereotypes and tagged values makes the diagram enigmatic and obscure. In a couple of situations, simple and intuitive concepts can be complicated by many diagrams, describing the system in various levels of abstraction.

In this paper I propose the UML notation extensions that can be helpful in modeling the multi channel access information system. The additions focus on the user's interaction with the system. The main goal of the work is to support the system analysts with new tools enabling concentration on the user and his relations to the modeled system.

2.1. Definitions

Channel – the route of communication or access. Channel is specified by the client access point (source), system access point (destination) and access method (via http, WAP or customized application).

Triggered use case – the use case that is fired by execution of other use cases.

2.2. Client channel

User classification is the general concept in the CRM IS [1,2]. Each type of the user can access one of the integrated systems (destination) via different access methods. Depending on the channel used to connect to the system, the user has a limited (or broadened) number of functionalities available offered by the software. Neglecting the limitations reasons, the importance of that fact has to be pointed out during the system modeling. For the systems that offer multiple channel access methods I propose the symbol of actor-channel. It represents the mixture of the classified user of a given type and the method it accesses the systems (Fig.1).
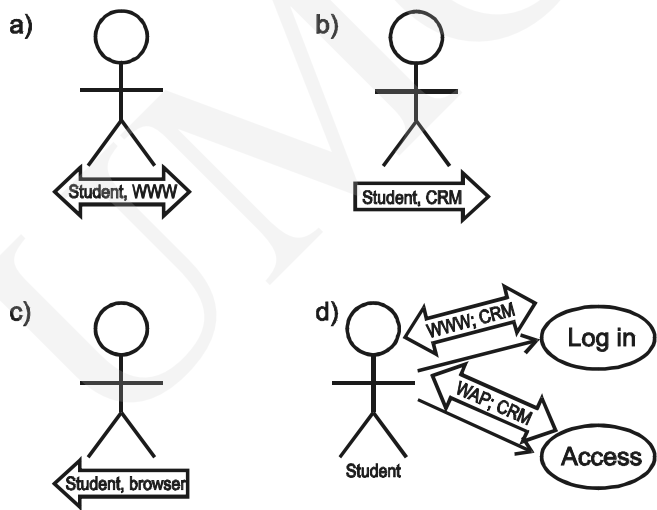

Fig. 1. Symbols representing the actor that uses certain access channel. a) the actor with the channel used represents the user of one possible communication channel (WWW, SMS, WAP) to access the information system. b) the actor and the destination system used when the information about channel is not necessary to describe the system c) the actor with the usage source. The symbol is equivalent to "client unit" proposed by Lin and Lee [5]. Mixture of classified user and categorized execution environment. d) Exemplary analogous description of the user channel and the use cases related to the actor. The channel (source, destination) is described by the arrow connecting the actor and the use case

These extensions give a new opportunity for modeling users interaction with the system. A person that models the user-system interaction can focus only on a given channel or one of its ends (source or destination). The notation patterns are presented in Fig. 2.
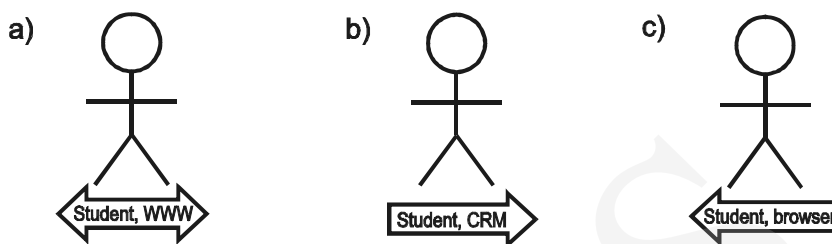
Fig. 2. Notation patterns. The actors are related with the channel they use the system. Three actors represent. The Student that users the web channel, the Student that uses the CRM system (no information about the access method or client software) and the Student that users the web browser (no information about the system)

In most cases usage if the icons is equivalent. For example, accessing the CRM system by the student via HTTP involves web browser usage. But if is also possible to access some part of the system via HTTP without the web browser. This situation takes place if the source of the channel is standalone application that uses WebServices technology. The channel is still HTTP but the channel source changes. If the functionalities offered by the web browser are different from those offered by the heavyweight application, these extensions allow to show them explicitly.

Secondly, the proposed notations allow to model the system in a more general way. It is possible to model overall functionalities available for the student using the CRM system and then focus on the functionalities offered to the user accessing the system by particular channel or even the client software.

2.3. Triggered use cases

Typically, use cases describe overall functionalities offered by certain system to a given user. If there are multiple cooperating information systems, it often happens that the system exchanges the information with each other. The information transfer or synchronization can be started periodically (regularly or called by administrator). Sometimes the process of sending information to other systems can be fired by user's activity or action. The triggered use cases (TrUC) are a type of use cases that are not utilized by the actor directly. Execution of those use cases is fired by the user's action. There is no need, for the UC execution environment (information system affected by the UC) to be the same systems that user currently interacts. Moreover, the pattern does not describe the technical details how the TrUC is realized. For example, login on the one web page fires login UC in the other system (Fig. 2). This function aims at avoiding necessity of double authentication when changing the system. From the technical point of view, this feature can be realized by setting the domain cookies, calling remote bean (in Enterprise Java Bean technology) or using SOAP protocol. The

decision is left for further levels of analysis. Nevertheless, the TrUC are additionally described by the information about systems affected by them.
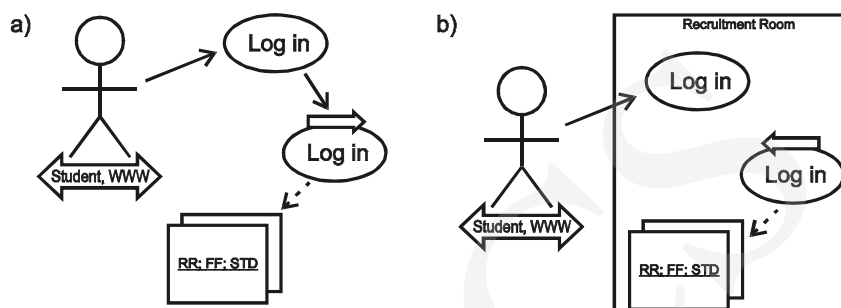

Fig. 3. Symbols of triggered use cases (TrUC). Similar to the user's channel TrUC are marked with the arrow above the UC use case ellipsis. Such use cases are not used by the actor directly. They are triggered after (or before) the use of "traditional" use case. Usually, execution of the TrUC affects and runs on different module or system that calling UC is "executed". In such a situation we can add additional information (double rectangle) that notifies, which modules of the system the UC affects. a) The Student communicating through WWW logs into one of the integrated system. Successful login fires login UC, that automatically logs the user to three other, independent systems (RR; FF and STD). b) While modeling with the use case diagram, we can also put TrUC that are triggered (arrow to the left) by some actions on one of the mentioned systems

The triggered use cases can be applied in the situations when we want to avoid or hide business mechanisms realization of particular process. The TrUC allows also to obscure technical aspects of use case realization. For example, if the system offers different, independent subsystems, successful authorization of one of them can affect the others. When the user logs into the system it calls indirectly the other use case that enables the access to the other independent systems. It gives the user the possibility to enter another system without filling the authorization form. Nevertheless, the technological aspects and the way of realization of the functionality are hidden. From a technical point of view, it can be realized using several methods (setting domain cookie or calling the remote methods). These aspects are not important in this level of modeling system, so they can be omitted. They also depend and are limited to technology chosen to implement the systems. The TrUC allow to leave the consideration of these aspects (in this stage of system modeling) and only points out to the functionality offered by the system.

2.4. Activity diagram and use case selection

The activity diagrams can be used to describe workflows that take place in the system. The actor channels and triggered use cases introduced in the previous sections aim at customizing functionalities and behavior depends on certain channel. The notation and semantics of activity diagram obviously do not

support those features. The proposed extension to activity diagram aims at supporting use cases filtering and workflow customization depends on channel, source or destination. Figure 4 presents the exemplary use case selection. The use case filtering can be performed according to channel, its destination, source or even actor himself. The use cases chat passes of the filter (<P>) are avaliable to use by the user connection via certain channel. The others are ignored (<I>). They can be totally unavailable or disabled, but visible and presented on the user's interface.
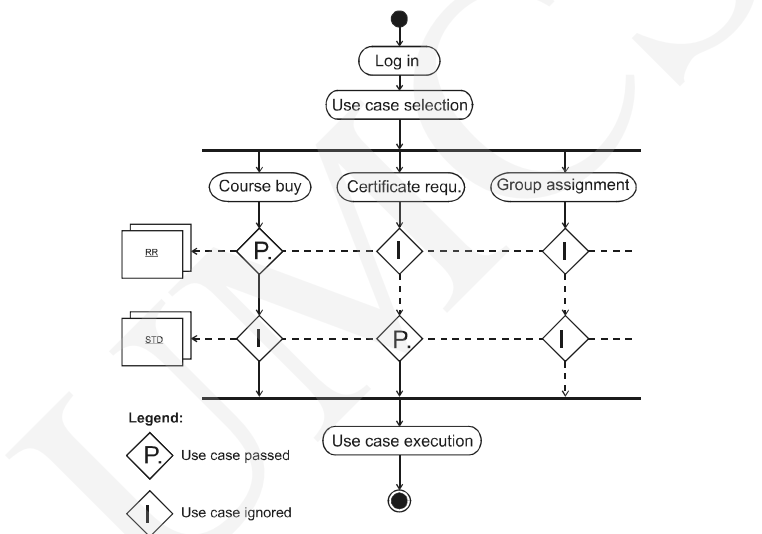

Fig. 4. Modified activity diagram with the proposed extensions. The diagram presents use case filtering possibility. Filtering can be using channel (or its prat: source or destination). The figure presents which of the functionalities are available for the user in the specific system (channel destination). The condition <P> passes the functionality to be able to execute. The condition <I> disable using a given functionality

This kind of diagram can be used for overall description of the user's interface. Ignored functionalities may be not visible or can be disabled for the user. The traditional UML activity diagrams can be related to certain use case. Their goal is to describe the data and activity workflow using the system. The proposed extensions need not be related to any use case. They sketch the availability of functionalities for the channel.

Conclusions

This work proposes the extended UML notations for the user oriented, multi-channel access and integrated information systems. Its goal is to alleviate the lack of modeling method for the analysis of multiple, incorporated IS. The user oriented modeling approach gives opportunity to describe the user aware

software such as CRM IS. Based on CRM concepts, My method introduces and focuses on notations for describing possible behavior of actors communicating through a given channel. The concept includes the situation where there is a number of integrated systems that communicate. This communication depends on the user's activity. Moreover, the new notation can model the system on high level of abstraction, neglecting the technical details of implementation. The possibility of selecting functionalities based on the user's channel is critical for the analysis method that makes the reliable CRM IS truly satisfy customers' requirements.

The UML 2.0 introduced the definition of the ports [11]. This new feature points out to the necessity of differentiation of the ways of accessing and usage software. Nevertheless, it can be applied only while modeling the structural aspects of the system. The extensions proposed in this paper fulfil similar needs while modeling the behavioral features.

These extensions, as many other developers by researchers are hard to apply in the analysis process. The notations proposed in the work need to be implemented for software modeling tools. Therefore, nowadays only the UML stereotypes give the reliable way of extending UML. Nevertheless, stereotypes as the method of limiting the meaning of symbols do not involve the semantics of the pattern.

References

[1] Cingil I., Dogac A., Azgin A., *A boarder approach to personalization*, CACM, 42(8) (2000) 136.
[2] Kramer J., Noronha S., Vrego J., *A user-centered design approach to personalization*, CACM, 43(8) (2000) 45.
[3] Mobasher B., Cooley R., Srivastava J., *Automatic personalization based on web usage mining*, CACM, 43(8) (2000) 142.
[4] Mulvenna M., Anand S., Buchner A., *Personalization on the net using web mining*, CACM, 43(8) (2000) 123.
[5] Ba S., Kalkota R., Whinston A., *Using client-broker-server architecture for Intranet decision support*, Decision Support Systems, (19) (1997) 171.
[6] O'Keefe R., Mceachern T., *Web-based customer decision support systems*, CACM, 41(3) (1998) 71.
[7] Lin J., Ch. Lee M., *An object-oriented analysis method for customer relationship management information systems*, Information and Software Technology, (46) (2004) 433.
[8] Jackobson I., Booch G., Rumbaugh J., *The unified software development process*, Addison Wesley, Reading, (1999).
[9] Jackobson I., Booch G., Rumbaugh J., *The unified modeling language reference manual*, Addison Wesley, Reading, (1999).
[10] Booch G., Jackobson I., Rumbaugh J., *The Unified Modeling Language User Guide*, Addison Wesley, Reading, (2002).
[11] Rumbaugh J., Jackobson I., Booch G., *The Unified Modeling Language Reference Manual*, Second Edition, Addison Wesley, (2004).