International Journal of Electrical and Computer Engineering (IJECE)
Vol. 10, No. 2, April 2020, pp. 1980~1992
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i2.pp1980-1992

Designing and modeling of a multi-agent adaptive learning system (MAALS) using incremental hybrid case-based reasoning (IHCBR)

El Ghouch Nihad, Kouissi Mohamed, En-Naimi El Mokhtar
LIST Laboratory, the Faculty of Sciences and Technologies, Abdelmalek Essaadi University, Morocco

Article Info

Article history:

Received Jun 1, 2019
Revised Sep 30, 2019
Accepted Oct 30, 2019

Keywords:

Adaptive learning system
Personalized path
Felder-Silverman learning style model (FSLSM)
Incremental hybrid case-based reasoning (IHCBR)
Machine learning
Multi-agent system (MAS)

ABSTRACT

Several researches in the field of adaptive learning systems has developed systems and techniques to guide the learner and reduce cognitive overload, making learning adaptation essential to better understand preferences, the constraints and learning habits of the learner. Thus, it is particularly advisable to propose online learning systems that are able to collect and detect information describing the learning process in an automatic and deductive way, and to rely on this information to follow the learner in real time and offer him training according to his dynamic learning pace. This article proposes a multi-agent adaptive learning system to make a real decision based on a current learning situation. This decision will be made by performing a hypride cycle of the Case-Based Reasonning approach in order to follow the learner and provide him with an individualized learning path according to Felder Silverman learning style model and his learning traces to predict his future learning status.To ensure this decision, we assign at each stage of the Incremental Hybrid Case-Based Reasoning at least one active agent performing a particular task and a broker agent that collaborates between the different agents in the system.

Corresponding Author:

El Ghouch Nihad,
LIST Laboratory, Department of Computer Science,
Faculty of Sciences and Technologies,
Abdelmalek Essaadi University,
Avenue Khenifra, Tétouan 93000, Morocco.
Email: nihad_elghouch@hotmail.fr

1. INTRODUCTION

The emergence of Web 2.0 has contributed to the development of adaptive learning by broadening its interaction possibilities, such as personalization by involving a wide variety of technologies derived from artificial intelligence. Adaptive learning systems aim to provide the necessary pedagogical tools to create a climate conducive to learning among learners. They need to adapt learning to the preferences, needs and learning styles of each learner to ensure real-time, personalized follow-up during the learning process, they must also integrate many actors with heterogeneous knowledge, acting with autonomy and cooperating with each other to succeed in the learning process. Therefore, these systems are recently based on artificial intelligence technologies, allowing individualizing the learning path, improving the effectiveness of training and avoiding failures and demotivation. In order to improve the quality of the adaptation and to instantly take into account the arrival of new data during the learning process, as well as to allow the reuse of existing experiences, we are oriented to use the Incremental Hybrid Case-Based Reasoning approach (IHCBR) which makes the right decision in real time, using both the execution of a classical cycle (linear process) or a dynamic cycle (dynamic process) of the Case-Base Reasoning [1] and to integrate the concept of agent [2] which presents autonomous entities that can interact intelligently in an environment to lead to a specific task.

In this article, we introduce the concept of intelligent agent in the field of e-learning and we present the architecture of a multi-agent adaptive learning system allowing to follow the learner in real time and to predict his future learning. The rest of this article is organized as follows: the second section is devoted to the presentation of the Case-Based Reasoning (CBR) approach and Multi-Agent Systems (MAS) [3]. The third section is devoted to the presentation of our architecture of an adaptive multi-agent learning system based on the IHCBR. In the fourth section, we present the operation of our architecture by modeling the different tasks performed by the system agents, using the Agent Unified Modeling Language (AUML) [4].

2. STATE OF THE ART

2.1. Case-based reasoning

Case-Based Reasoning (CBR) is a paradigm of artificial intelligence that solves a problem by relying on the solutions of past experiences already solved [1], using a classical cycle that includes the following steps: Elaboration, Retrieve, Reuse, Revise and Retain. These steps are detailed in [5-7]. There is an adaptation of the classical CBR cycle with a change of order and the content of the steps, it is the Incremental Dynamic Case-Based Reasoning (IDCBR) proposed by [6, 7], whose cycle is continuous and it takes into consideration the dynamic and incremental change of the descriptors describing the problem to be solved. This new cycle allows to stop the execution of certain steps and to re-execute others at each moment, there is a change of these descriptors. The cycle is shown in Figure 1.

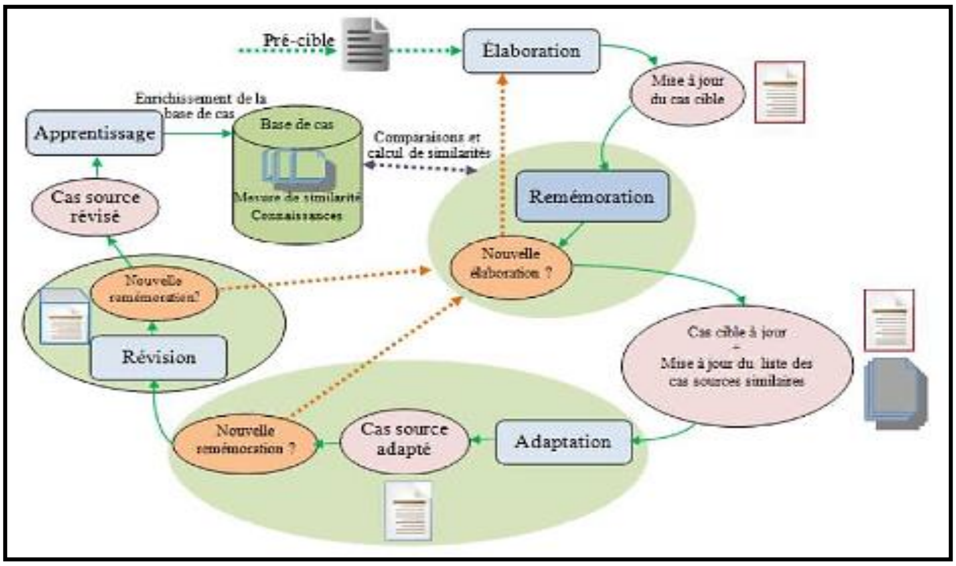

Figure 1. IDCBR cycle [6, 7]

2.2. Multi-agents systems

According to [2], an agent: "is a computer system located in a certain environment, capable of exercising autonomously actions on that environment in order to achieve its objectives". A Multi-Agent System (MSA) is one of the research axes in Distributed Artificial Intelligence (IAD), it contains a set of agents that interact with communication protocols and they are able to act on their environment. Multi-Agent Systems are composed of [3]:

- An environment that has a space having usually a metric.
- A set of objects located in the environment, these objects can be perceived, created, destroyed and modified by agents.
- A set of agents that represent the active entities of a system.
- A set of relationships that link agents together.
- A set of operations that allows agents to perceive, produce, consume, transform and manipulate objects.
- A set of operators who are responsible for representing the application of these operations and the reaction of the environment towards the modifications.

Adaptive learning systems using agents must benefit from the advantages provided by agent technology such as autonomy, flexibility and communicability. Each agent represents the different elements that make up the adaptive learning systems (learner, tutor, expert, etc.) and assures a particular role [8].

3. PROPOSED ARCHITECTURE IHCBR

3.1. Presensation of the archiecture

To improve the quality of adaptation, test the effectiveness of our proposed architecture in [9] and take advantage of the benefits offered by agents such as autonomy, flexibility, communicability and distributed problem solving, we propose an architecture of a multi-agent adaptive learning system based on Incremental Hybrid Case-Base Reasoning (IHCBR), whose reasoning cycle combines the classical cycle (the order of the steps is important, there is no change during the execution of the cycle) and the dynamic cycle (the order does not matter, a detected change causes the re-execution of some steps several times).

Our proposed architecture offers a personalized learning path dynamically in real time, based on the Felder-Silverman Learning Style Model (FSLSM) [10], the dynamic and incremental change in learner behavior during the learning process and previous successful experiences of other learners by associating an independent intelligent agent at each step of the IHCBR and ensuring communication between these agents to perform real-time monitoring and prediction tasks of a learning situation dynamic. Our architecture as shown in Figure 2 includes a set of agents who collaborate with each other to assist the learner in real time who has encountered a learning problem by providing individualized learning and predicting their dynamic learning situation. These agents perform specific roles and are assigned to each step of the IHCBR process.

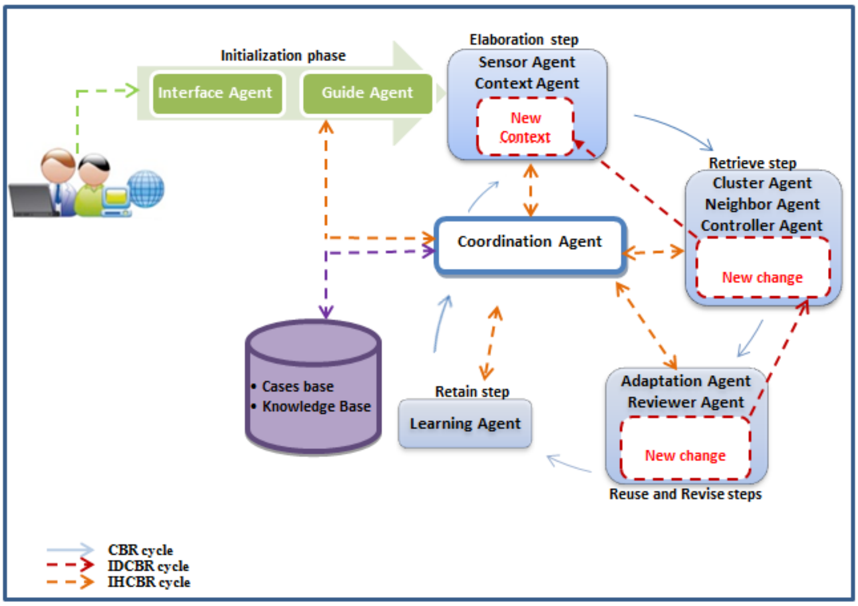

Figure 2. Our proposed architecture

To improve the quality of the adaptation and to test the efficiency of our proposed architecture and the advantages offered by the taechnology agent such as autonomy, flexibility, communicability and distributed problem solving, we propose an architecture of a Multi-Agent Adaptive Learning System based on Incremental Hybrid Case-Base Reasoning (IHCBR), whose reasoning cycle combines the classical cycle (the order of the steps is important, there is no change during the execution of the cycle) and the dynamic cycle (the order does not matter, some steps can be run multiple times).

Our proposed architecture offers a personalized learning path in real time, based on:

- The Felder-Silverman Learning Style Model (FSLSM) [10] ;
- The dynamic and progressive change in learner behavior during the learning process;
- The previous successful learning experiences of other learner;
- The association of intelligent and independent agents at each stage of the IHCBR by providing real-time communication, monitoring and forecasting of a dynamic learning situation.

Our architecture as shown in Figure 2 includes a set of agents who collaborate with each other to assist the learner in real time who has encountered a learning problem by providing individualized learning and predicting their dynamic learning situation. These agents perform specific roles and are assigned to each step of the IHCBR process. The Table 1 shows the distribution of the agents at the different steps of IHCBR, as well as the tasks performed:

Table 1. Agents of the system and their tasks in the different steps of IHCBR

Phase/Step	Agent	Task
Initialization phase	Interface Agent Guide Agent	Collect data from learners Suggest an initial learning path according to the learning style of FSLSM.
Acquisition phase		
– Elaboration step	Sensor Agent Context Agent	Collect learning traces; Detect the learning problem Make the right decision.
– Retrieve step	Cluster Agent Neighbor Agent Controller Agent	Create a list containing the experiences closest to the current learning situation that can be modified following a change in the situation.
– Reuse and Revise step	Adaptation Agent Revise Agent	Evaluate the resemblance between the selected past experiences and the current learning situation that can be modified following a change in the situation
– Retain step	Learning Agent	Report that the problem is resolved and that it becomes a new experience to back up.
Knowledge Base	Coordination Agent	Save and update all information sent by other agents; Ensure communication between the different agents.

3.2. Description of the architecture

3.2.1. Interface agent

The Interface Agent serves to manage the communication between the learners and the adaptive multi-agent learning system. It allows to:

- Authenticate new learners;
- Identify learners;
- Detect the profile of the learner expressed by the FSLSM Learning Style Test;
- Create and distribute mobile agents on mobile devices, if the learner is connected by a mobile device using the http protocol.

In our architecture, we use the Felder-Silverman Learning Style Model (FSLSM) to detect the learning style of each learner for several reasons given in [11-13].

3.2.2. Guide agent

The Guiding Agent aims at displaying and proposing a learning path based on the learner's learning style, relying on the correspondence between the FSLSM learning style and the metadata describing the units of the course generated by eXeLearning software [14] based on differentiated pedagogy. According to [15-18], there is a relationship between the educational resources and the learning styles of the FSLSM. If the learner is connected by a mobile device, this agent becomes a mobile agent that offers a course based on the FSLSM learning style and the characteristics of the mobile devices detected by the Sensor Agent.

3.2.3. Sensor agent

The Sensor Agent is a reactive agent, making it possible to describe the learning environment at each instant ti of the learning process. This agent makes it possible to create the learning traces of each learner at each moment $_{ti}$ which contains a set of indicators that we considered relevant presented in the form of a vector named $Episode_{(ti,j)}$ describing a trace left by the learner j at the moment t_i:

$$Episode_{(ti,j)} = \begin{pmatrix} t_i \\ Crs \\ VOb \\ DtVOb \\ NbrVstVOb \\ DVstVOb \\ FVOb \\ RVOb \end{pmatrix}$$

- t_i: episode t_i corresponds to the instant t_i
- Crs: chosen course;
- VOb: learning object;
- DtVOb: date of the visit of the learning object;
- NbrVstVOb: number of visits of the learning object;
- DVstVOb: duration of the visit of the learning object;
- RVOb: resource of the learning object;
- FVOb: nature of the learning object.

This agent is activated as soon as the learner starts to learn. It is responsible for creating the vector $Episode_{(ti,j)}$ at each episode t_i of the learning process and sending them to the Coordinator Agent to save them in the knowledge base. If the learner is connected by a mobile device, the Sensor Agent becomes a reactive mobile agent that also collects the characteristics of the mobile device as a vector $Mobile_{(ti,j)}$:

$$Mobile_{(ti,j)} = \begin{pmatrix} t_i \\ T_{apM} \\ BP_{apM} \\ B_{apM} \end{pmatrix}$$

- t_i : episode t_i corresponds to the instant t_i
- T_{apM} : memory size;
- BP_{apM} : bandwidth;
- B_{apM} : battery level

3.2.4. Context agent

The Context Agent will make the appropriate decision based on the information received by the Sensor Agent, by observing and analyzing the observed behavior of the learner. The Context Agent determines whether this is a normal or abnormal learning process, filtering and sorting the information about the current learning environment so that it can be handled according to the type of problem detected in the current learning environment at every change in the behavior of the learner.

3.2.5. Cluster agent

Using the unsupervised machine learning method Fuzzy C-Means (FCM) [19-21], the Cluster Agent classifies learners into homogeneous clusters at the level of their behaviors (the learning traces left behind in the system and the characteristics of the mobile device) at each moment t_i, in order to identify the cluster that belongs the learner who has encountered a learning problem in the form of a vector V_Clus_{ti}. The purpose of using the FCM method is to reduce the sample size and the learners' classification time.

3.2.6. Neighbor agent

The Neighbor Agent creates a matrix M_Clus_{ti}, each line of this matrix contains the vector $Episode_{(ti,j)}$ (j belongs to V_Clus_{ti}) and applying the K-Nearest Neighbor algorithm (K-NN) [22-25], we get the list of the K nearest learners at each moment t_i by calculating the similarity between each row of the M_Clus_{ti} matrix and the current situation, the similarity values are stored in a correspondence table. Since the learner's behavior evolves dynamically over time (the arrival of new data), the list retrieved can be updated with each evolution.

3.2.7. Controller agent

The Controller agent is listening for a new detected change. In this case, it sends a message to the Cluster Agent who in turn sends it to the Neighbor Agent to update the list of nearest candidate learners.

3.2.8. Adaptation agent

The Adaptation Agent is responsible for validating the experiences of the nearest prospective learners, either by choosing the first experience offered, or by giving the tutor the opportunity to choose one of the proposed experiences.

3.2.9. Reviewer agent

The Reviewer Agent is listening for a new detected change. In the case of change, there will be a new elaboration of the current learning situation and an update of the list of nearest similars learners, leading to a modification of the correspondence table.

3.2.10. Learning agent

The Learning Agent is responsible for presenting the current learning situation and its solution as a new future experience.

3.2.11. Coordination agent (brocker agent)

All information circulated in our architecture is recorded in the knowledge base (case base, correspondence table, ...). This database is managed by the Coordination Agent who also allows the communication and the synchronization between the different agents to ensure the performance of the system. He plays the role of a broker by allowing to:

- Save the personal information and the profile of the learner;
- Save the learning traces at every moment t_i;
- Update the current context of learning at each change;
- Save the learning path;
- Update the learner profile;
- Send requests to the agents concerned.

4. DESIGNING AND MODELING OF OUR PROPOSED ARCHITECTURE (IHCBR)

To present the functioning of our system, we use the Agent Unified Modeling Language (AUML) [4] modeling which presents a graphic modeling language based on UML. AUML is a support notation for agent-oriented development systems, it provides several types of representation covering system description, components, system dynamics and deployment. Multi-Agent system designers already use AUML to represent agents, their behaviors, and their interactions with each other [26, 27]. The advantage of the AUML language is that it is for Object-Oriented Programming (OOP), which allows developers to easily move from object-oriented methodology to the agent-oriented approach. In this part we describe the scenarios performed by the different agents of the system, by representing their behaviors through the use of class diagram, use case diagram and sequence diagrams.

4.1. Use case diagram

A use case diagram is a feature of the system, it includes a family of scenarios where each scenario is a special treatment of the system as shown in Figure 3 [28].

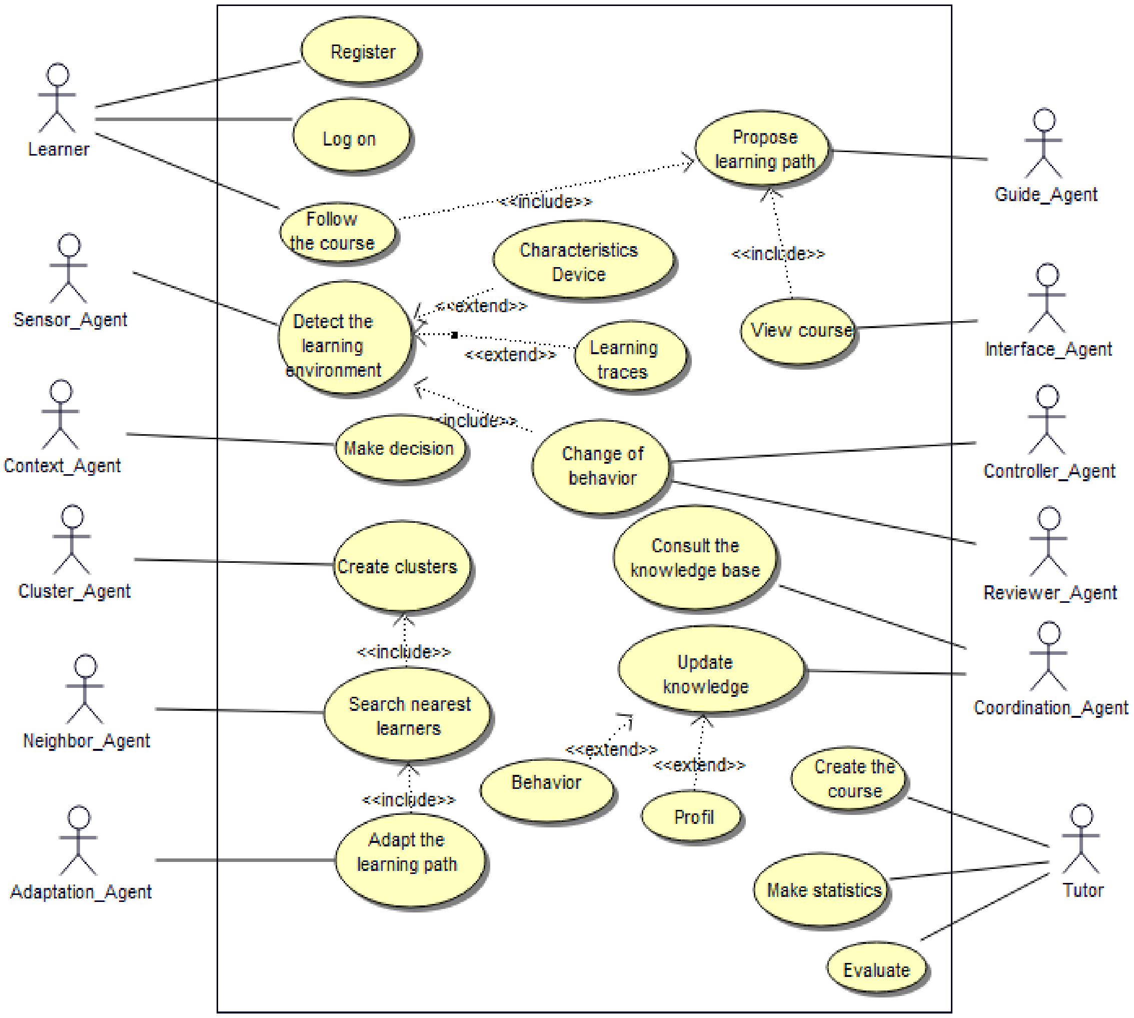

Figure 3. Use case diagram of system global

4.2. Class diagram

Our class diagram is designed in two layers:
- The first layer presents agents and their interactions with the environment;
- The second layer presents the interactions between the differents actors of the system.
- The first layer of the class diagram as shown in Figure 4, it contains the following classes:

- Agent: it presents the main class of the diagram allowing to describe the agent and its properties (roles, attributes and perception);
- Environment: it groups all the environmental data (attributes and perceptions);
- Interaction: it presents a class of reflexive association between the agents making it possible to record the actions carried out between the agents;
- Action: it presents an association class between the agent and the environment allowing to record the messages exchanged between the agents and to list all the possible actions that an agent can execute on its environment;
- Behavior: it is used to model several tasks that an agent can achieve;
- Reactive_Agent: It has a type of agent that has the same properties of the Agent class;
- Cognitive_Agent: it presents a type of agent, where the agent can decide to execute an action;
- Hybrid_Agent: it extends from the Agent class, it represents a hybrid agent behavior that is a compromise between the two reactive and cognitive agents;
- BDI_Agent: It extends from the Cognitive Agent class.

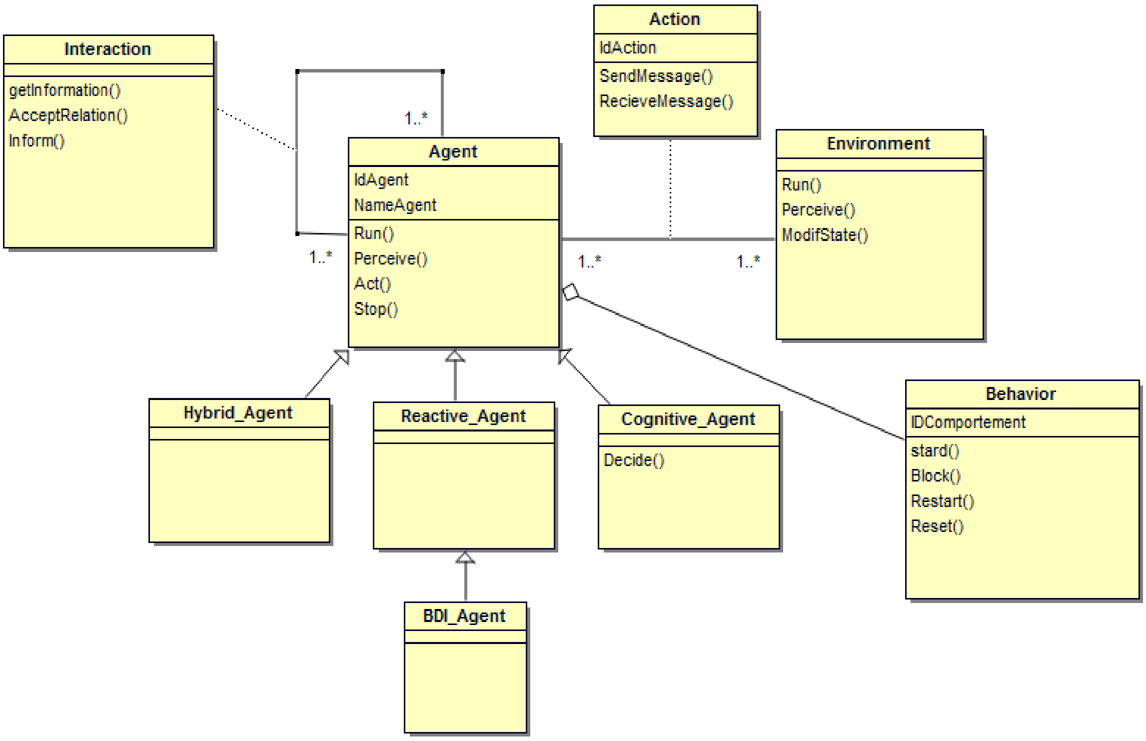

Figure 4. First layer of the generic class diagram for SMA

The second layer of class diagram as shown in Figure 5, it contains the following classes:

- Learner: it contains all the personal information describing the learner;
- Learning_style: contains the result of the FSLSM Learning Style Model test;
- Learning_Traces: it contains all the information describing a trace left in the system;
- Device_Character: it contains all the information describing the mobile device;
- Learning_Path: it contains the learning path proposed and validated;
- Learning_Object: contains all the information to describe a learning object
- Version: It extends from the Learning_ Object class, adding version information (resource and format)Tutor: it contains information about the tutor;
- Evaluation: it contains all the information concerning the evaluation;
- Recommendation: it allows the decision to be made in case of a negative evaluation.

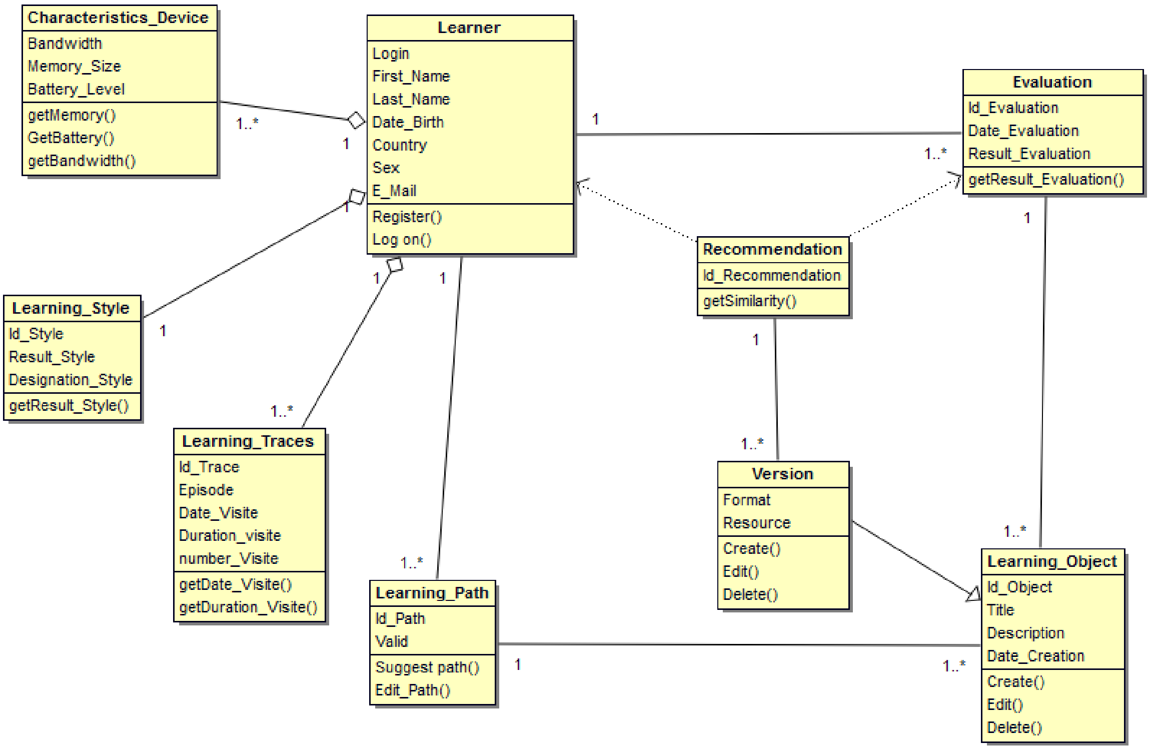

Figure 5. Second layer of the class diagram for SMA

4.3. Sequence Diagrams

The sequence diagram shows the message exchange between agents over time.

4.3.1. Scenario: "New Inscription"

The "new Inscription" scenario presents the communication between the learner and the Interface agent, which in turn communicates with the Coordination agent to validate the registration as shown in Figure 6. This scenario is done only once (the first contact of the learner with the system).

4.3.2. Scenario: "Connection"

The "Connection" scenario describes the process of authentication of learners by the Interface Agent that communicates with the Coordination Agent in order to validate the authentication details (login and password) as shown in Figure 7.

4.3.3. Scenario: "Beginning of Learning"

When the Interface agent validates the learner's authentication, the Guide agent is responsible for viewing the course by contacting the Coordination Agent to begin learning as shown in Figure 8.

4.3.4. Scenario: "Elaboration Step"

The "Elaboration Step" Figure 9 scenario describes the operation of the IHCBR Elaboration step which contains the Sensor agent and the Agent Context to make an appropriate decision at each instant t_i, based on the information collected by the Sensor Agent.

4.3.5. Scenario: "Retrieve Step"

The "Retrieve Step" Figure 10 scenario describes the operation of the IHCBR Retrieve step containing the Cluster Agent and the Neighbor Agent and the Control Agent to find learners with the most similar learning situations to the current situation, by contacting the Coordination agent.

4.3.6. Scenario: "Reuse and Revise Step"

The Figure 11 scenario "Reuse and Revise Step" describes the operation of the Reuse step and the Revise step of IHCBR (Adaptation agent and Reviewer agent), allowing adapting the learning situation according to the similarity realized in the Revise step

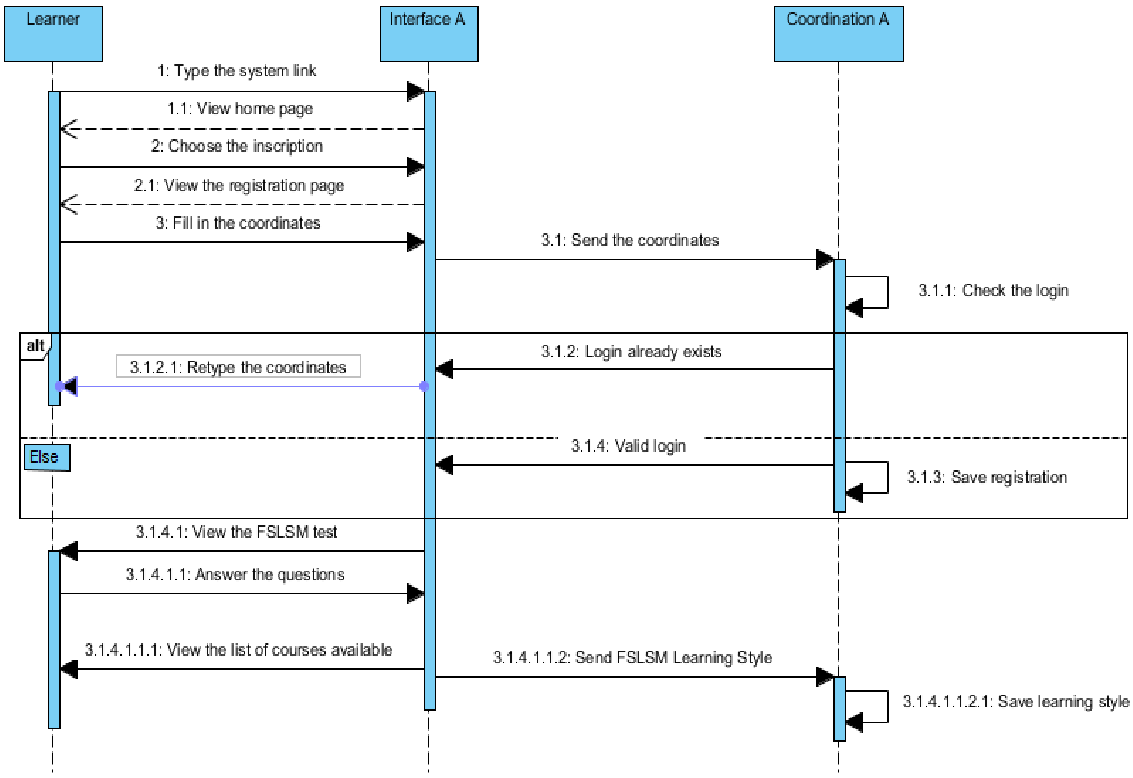

Figure 6. Sequence diagram of the scenario of "New Inscription"

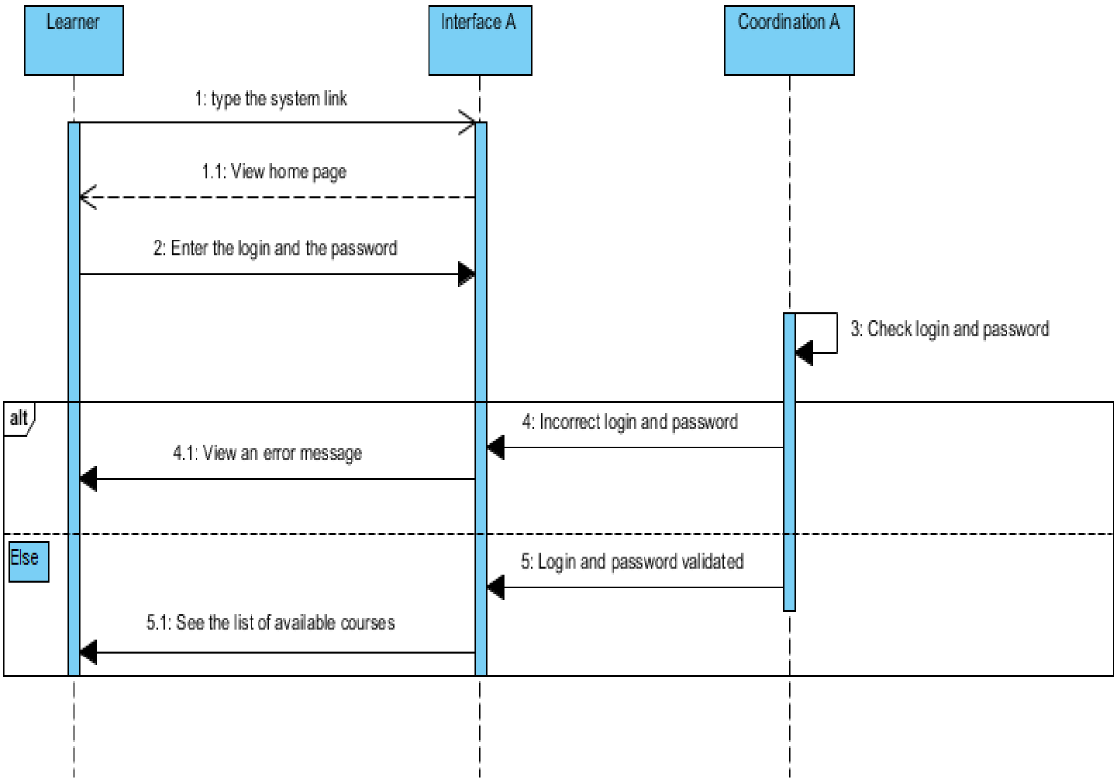

Figure 7. Sequence diagram of the scenario of "Connection"

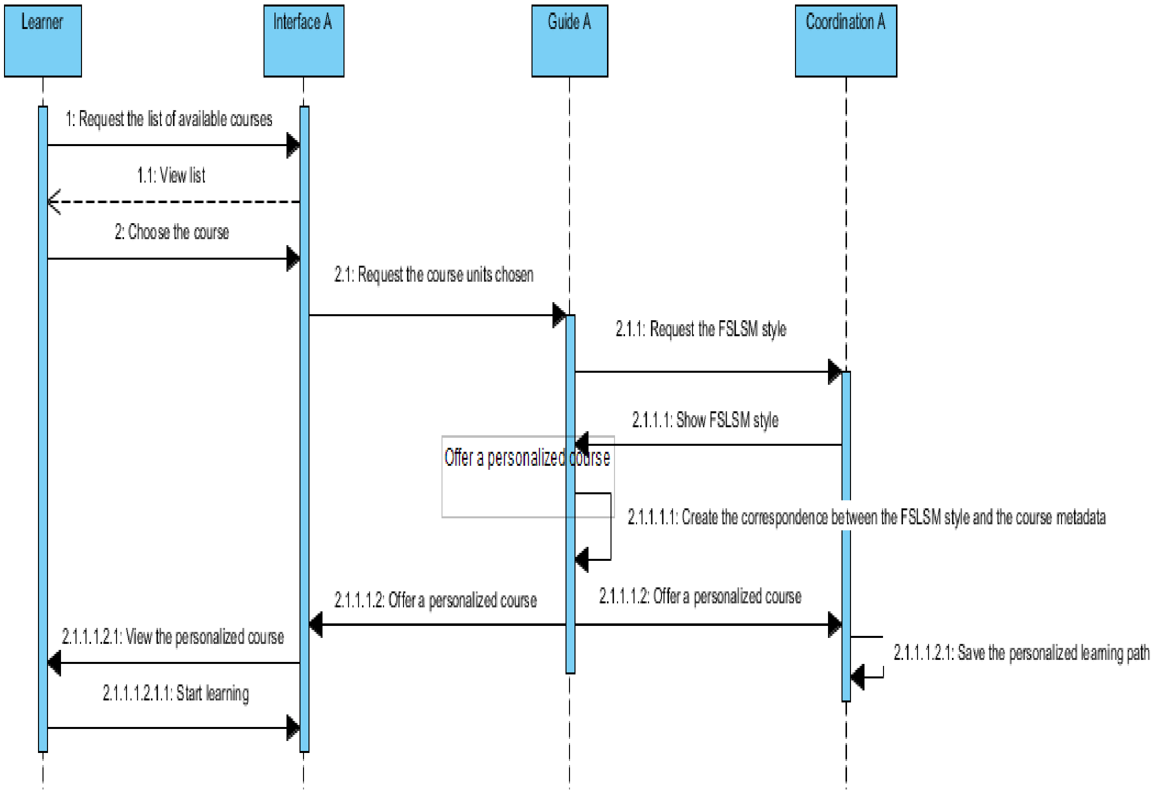

Figure 8. Sequence diagram of the scenario of "Beginning of Learning"

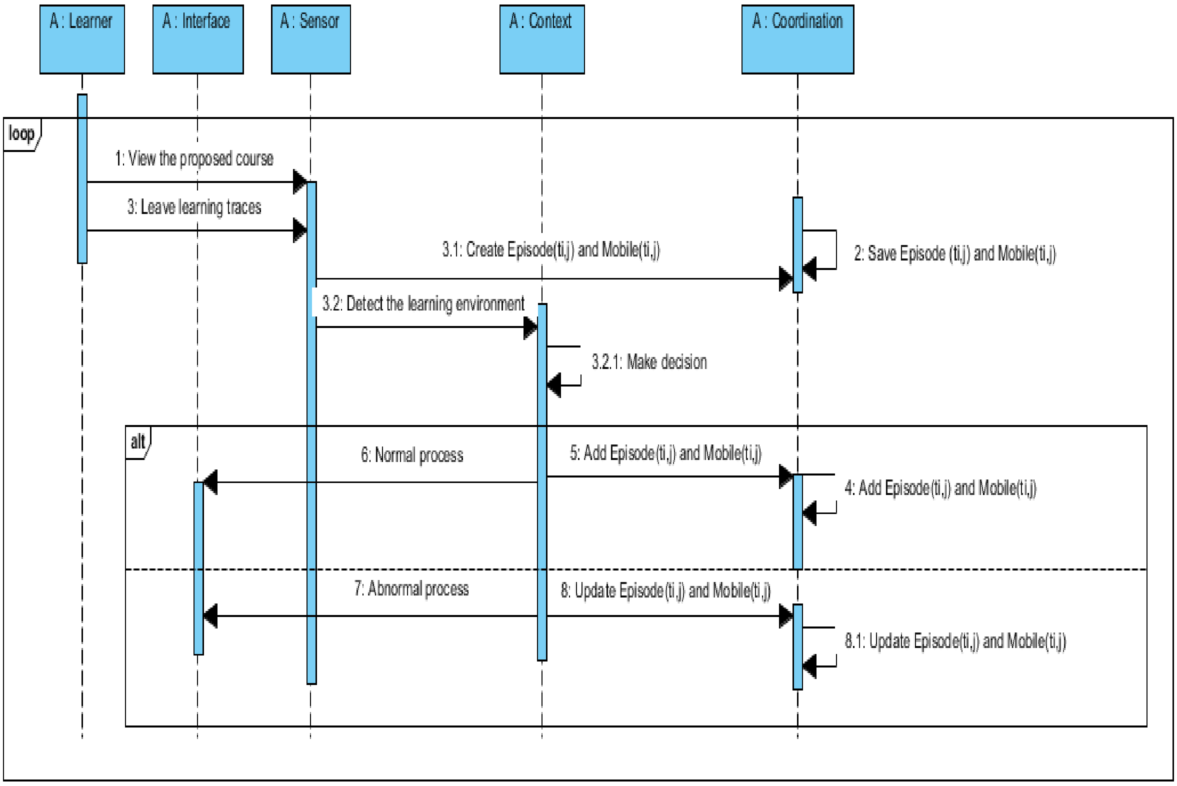

Figure 9. Sequence diagram of the scenario of "Elaboration Step"

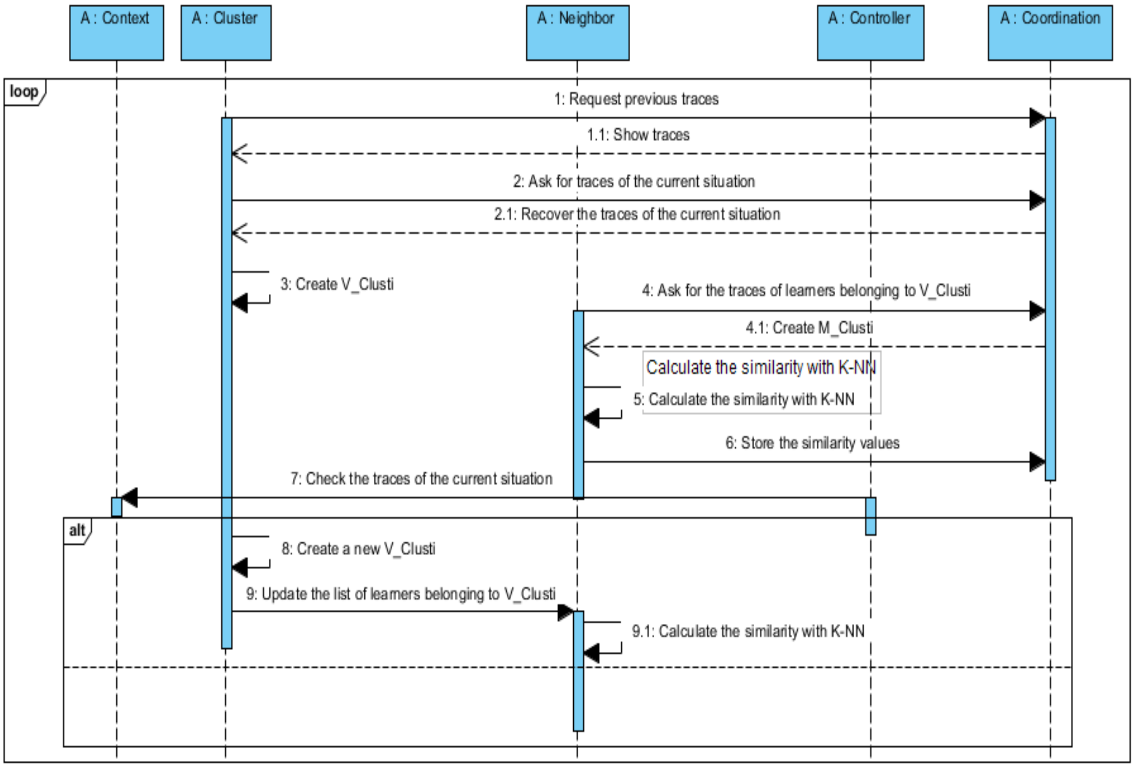

Figure 10. Sequence diagram of the scenario of "Retrieve Step"

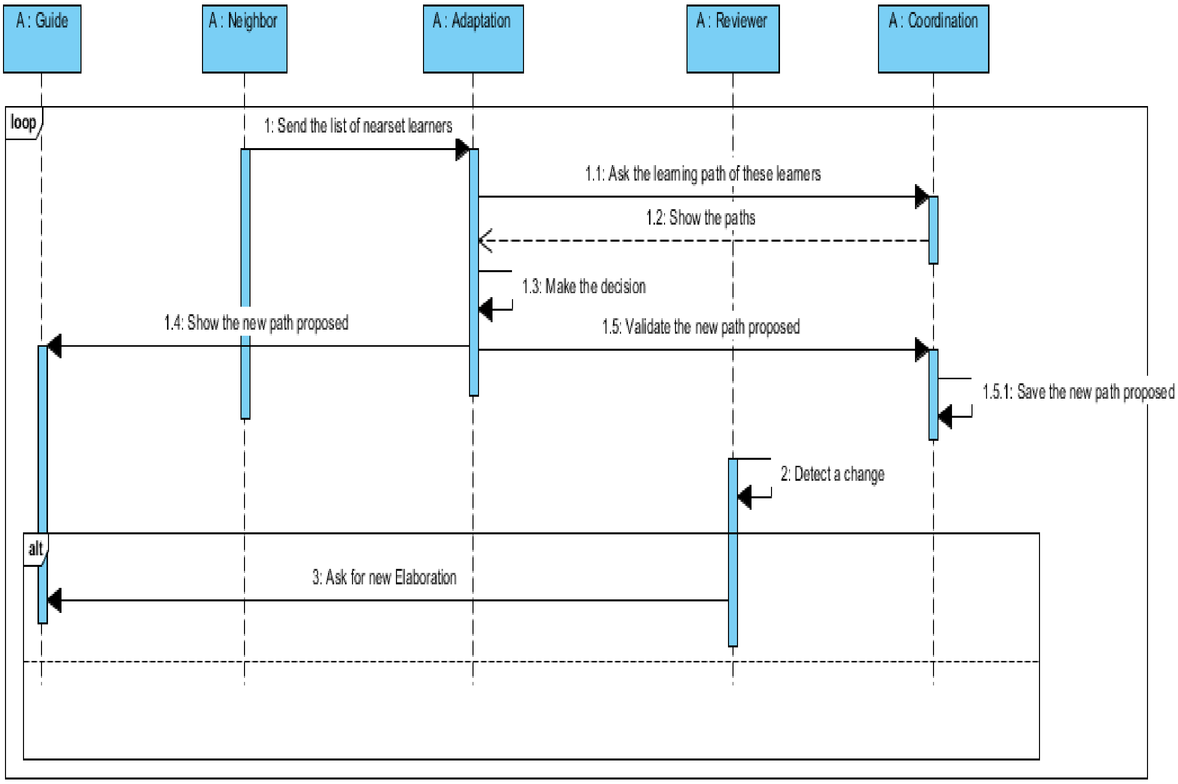

Figure 11. Sequence diagram of the scenario of "Reuse and revise Step"

5. WORK REPORT

In this paper, we propose an amelioration of an architecture of an adaptive learning system based on IDCBR proposed in [9] towards an architecture of adaptive learning Multi-Agents based on IHCBR. This amelioration is well illustrated in the Table 2.

Table 2. Amélioration made in the architecture.

Adaptive learning system based on IDCBR	Adaptive learning system Multi-Agents based on IHCBR
– Reasoning using a CBR dynamic cycle; – Use the K-NN method to find the nearest learners.	– Reasoning by using both a classical cycle (linear process) and a dynamic cycle (dynamic process) of the CBR; – Use the FCM method to decrease sample size, reduce learner classification time, and make it easier to find similar nearest learners using the K-NN algorithm; – Integrate agent technology with the benefits provided by agents (dynamicity, collaboration, autonomy, etc.).

6. CONCLUSION AND PERSPECTIVES

We propose adaptive learning system architecture to assist the learner in real time during the learning process by providing a course adapted to their learning environment. This architecture presents a multi-agent learning system, whose agents communicate via an incremental hybrid cycle of CBR, using a classical and dynamic cycle in the same learning process. This hybrid cycle integrates in the Retrieve step the FCM and K-NN methods to classify learners with similar behaviors into clusters to facilitate filtering ofof nearest learners to the current learning situation. Our system will make a adequate decision at each moment of the learning process based on the learner's profile, dynamic behavior change, device characteristics (if connected by mobile device), and previous experiences learners saved in the knowledge base in order to personalize their learning path and predict their future learning situation. To ensure this customization and prediction, we assigned at each step of the IHCBR at least one agent performing a particular task and collaborating with the agents of other steps through an AUML modeling. Our future work is the concrete implementation, in real cases, of the architecture of our Multi-Agent adaptive learning system under the JADE platform (Java Agent DEvelopment developed in Java. The choice of this platform is due to several reasons, such as its compatibility with most hardware and software configurations, its compliance with the FIPA standards and specifications (Foundation For Intelligent Physical Agents), it very precise architecture allowing the construction and execution of agents and its ability to communicate between JADE agents and non-JADE agents.

REFERENCES

[1] Kolodner J. L., "Case-Based Reasoning," *Morgan Kaufmann*, 1993.

[2] Michael Wooldridge, "On the sources of complexity in agent design," *Applied Artificial Intelligence*, vol. 14(7), pp. 623-644, 2000.

[3] Kaddour & Halima,"Mise en œuvre d'une application SMA Dans les réseaux P2P (supervision system)," Thèse de doctorat, Université Abou BakrBelkaid Tlemcen Faculté des Sciences, 2015.

[4] AUML: http://www.auml.org.

[5] Mille A., "Traces bases reasoning (TBR) definition, illustration and echoes with storytelling," Rapport Technique RR-LIRIS-2006-002, LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon1/Université Lumière Lyon 2/Ecole Centrale de Lyon, 2006.

[6] Abdelhamid Zouhair., "Raisonnement à partir de casdynamique multi-agentss: Application à un système de tuteur intelligent," PhD in computer science, in Cotutelle between the Faculty of Sciences and Technologies of Tangier (Morocco) and the University of Le Havre (France), 2014.

[7] El Mokhtar, En-naimi & Abdelhamid Zouhair, "Intelligent dynamic case-based reasoning using multiagents system in adaptive e-service, e-commerce and e-learning systems," *Int. J. of Knowledge and Learning (IJKL)*, vol. 11(1), pp. 42-57, 2016.

[8] Ronan Querrec., *"Les systèmes multi-agentss pour les environnements virtuels de formations,"* Application à la sécurité civile, thèse de doctorat Informatique, Laboratoire d'Informatique Industrielle (LI2/ENIB), Université de Bretagne Occidentale, 2002.

[9] El Ghouch N., En-Naimi E.M., Zouhair A., Al Achhab M., "*Individualized follow-up of the learner based on the k-nearest neighbors (K-NN) method embedded in the retrieval step of case based reasoning approach (CBR),"* In: Ben Ahmed M., Boudhir A., Younes A. (eds) Innovations in Smart Cities Applications Edition 2. SCA 2018. Lecture Notes in Intelligent Transportation and Infrastructure. Springer, Cham, 2019.

[10] Felder, R.M. & L.K. Silverman, "*Learning styles and teaching styles in engineering education*," Presented at the 1987 Annual Meeting of the American Institute of Chemical Engineers, New York, 1987.

[11] Kuljis J. & Liu F.A., "*Comparison of learning style theories on the suitability for elearning,"* In: *Hamza, M.H. (ed.)* Proceedings of the IastedConference on Web Technologies, Applications, and Services, pp. 191-197. ACTA Press, 2005.
[12] Sangineto E., Capuano N., Gaeta M., et Micarelli A., "Adaptive course generation through learning styles representation," *Journal of Universal Access in the Information Society*, vol. 7(1/2), pp. 1-23, 2007.
[13] Popescu E., "Dynamic adaptative hypermedia systems for e-learning," *Thèse de Doctorat,* Criova University, Romania, 2008.
[14] N. Elghouch, Y. Z. Seghroucheni, B. E. El Mohajir, M. Achhab, "*An application to index the didactic resources in an adaptive learning system (ALS),*" 5th International Conference on Information & Communication Technology and Accessibility (ICTA), pp. 1-3, 2015.
[15] Barbara A. Soloman & Richard M. Felder, "Index of learning styles questionnaire," NC State University, 2005. Available online at: http://www.engr.ncsu.edu/learningstyles/ilsweb. Html (last visited on 14.05. 2010).
[16] Sabine Graf, KinShuk, Tzu-Chien Liu., "Identifying learning styles in learning management systems by using indications from students' behavior," In *Advanced Learning Technologies*, 2008. *ICALT'08. Eighth IEEE International Conference on*, *IEEE*, pp. 482-486. 2008.
[17] Hong Hong & DrKinshuk., "Adaptation to studentlearning styles in web based educational systems," In *EdMedia: World Conference on Educational Media and Technology*, pp. 491-496, 2004.
[18] Elvira Popescu, Philippe Trigano, CostinBadica, *"Relations bet- weenlearning style and learnerbehavior in an educational hypermedia system: An exploratory study,"* In Advanced Learning Techno- logies, 2008. ICALT'08. Eighth IEEE International Conference on, IEEE, pp. 725-726, 2008.
[19] Fouad M. B., *"La logique floue pour classification des feuilles de vigne*," Mémoire Master Académique, Faculty of New Technologies of Information and Communication, University Kasdi Marbah, Ouargla, Algiria, 2015-2016.
[20] J. Hathaway, J. Bezdek, J. Davenport., "On relational versions of c-means algorithm," *Pattern Recognition Letters,* vol. 17, pp. 607-612, 1996.
[21] S. Miyamoto & K. Umayahara, "Methods in hard and fuzzy clustering," in Z.-Q. Liu, S. Miyamoto (Eds.), Soft Computing and Human-Centered Machines," *Springer*, pp. 85-129, 2000.
[22] Zhang Zhongheng, "Introduction to machine learning: k-nearest neighbors," *Annals of translational medicine*, vol. 4(11), 2016.
[23] Christophe Chesneau, "Éléments de classification," Master. France, 2016.
[24] Rémi Lajugie, http://remi-lajugie.fr/docs/kPpv.pdf, 2017.
[25] Beloufa Fayssal, "Conception d'un classifeur à base des règles floues," PhD in computer science, UniversityAboubekrBelkaid, Tlemcen, Algeria, 2016.
[26] J. Odell, H. V. D. Parunak and B. Bauer, "*Extending UML for agents*," In G. Wagner, Y. Lesperance, and E. Yu, editors, Proceedings of the Agent-Oriented Information Systems Workshop at the 17th National conference on Arti.cial Intelligence, Austin, Texas, 2000. ICue Publishing.
[27] B. Bauer, J. P. Müller, and J. Odell, "*An extension of UML by protocols for multiagent interaction*," In International Conference on MultiAgent Systems (ICMAS'00*)*, pp. 207–214, 2000.
[28] Mémoire en ligne, http://www.memoireonline.com : Conception et réalisation d'une application de gestion des marchés par appel d'offres au sein de l'entreprise Tunisienne d'Activité Pétrolière. Dernier accès mai 2016.