Computers and Education: Artificial Intelligence 3 (2022) 100057

Contents lists available at ScienceDirect

Computers and Education: Artificial Intelligence

journal homepage: www.sciencedirect.com/journal/computers-and-education-artificial-intelligence

Enhancement of online education system by using a multi-agent approach

Nethra Viswanathan, Sofia Meacham, Festus Fatai Adedoyin *

Department of Computing and Informatics, Bournemouth University, UK

ARTICLE INFO

Keywords:
Multi-agent system
Virtual learning environment
Online education
Simulation
Learning management systems

ABSTRACT

Multi-Agent System (MAS) is popular in the fields where cooperative effort is required to fulfil the purpose of the end product. This study creates a Multi-Agent System as a potential solution for the implementation of an online education system in order to bring about swift and accurate responses from the system and thus a meaningful conversation between the learners and the system. The existing work on online education system employing MAS does not depict the means of flow of information from the browser to the internal MAS and also lacks the adaptivity element with respect to the changing demands of the learners. The study attempts to bridge this gap by employing a system with Event-Condition-Action model and intelligent agents which include a message passing agent to depict the means of message flow from the browser to the MAS and an adaptive course organiser agent which organises adaptive educational content with respect to the changing needs of the learner. The outcome of the study is the design and simulation of the said system as a standalone entity that can be attached to a virtual learning environment (VLE), with the addition of pedagogical agents performing different functionalities to form an adaptive system. The system is evaluated by validating the results generated for various case studies using a validation tool. The expected behaviour is that the resulting course agenda agrees with the learning mode preference of the user provided initially and the results are expected to change according to the changing user preferences through a self-assessment questionnaire.

1. Introduction

This study provides a working prototype of the proposed method. Online education reduces learners' efforts to a great extent in finding their courses and discussing with like-minded groups with AI aided course guides and the availability of discussion forums (Neto, 2017). Such e-learning environments are seen on websites where tutors provide the course structure, content, forums, exams and such other classroom facilities to students online based on their curriculum as well as interests and preferences which keep updating with time, their learning level and time dedicated for each resource (Neto, 2017). The paper focuses on improving the adaptivity and thus the overall user experience of such e-learning environments with the help of artificial intelligence. There has been increasing interest in the uses of technology among teachers and educators since the early 2000s through face-to-face and online instructions for course teaching (Chen, Zou, Cheng, & Xie, 2020). Applications and tools driven by AI technologies, for instance, intelligent robots and adaptive learning systems, have been increasingly utilized by educators and learners within both K-12 and university contexts. AI technologies provide opportunities for the realization of personalized learning for learners to meet their individual needs (Chen, Xie, Zou, & Hwang, 2020). Personalisation and adaptivity are the key factors considered by users in online education given the presence of the numerous choices on the internet (Nadrljanski, Vukic, & Nadrljanski, 2018). Hence the paper focuses on the addition of adaptivity features to an existing virtual learning environment (VLE) which customises the learning content of every user based on his/her learning mode preferences. The system is proposed to be designed and simulated as a stand-alone prototype which can serve as an extension to a VLE.

MASs function in a cooperative fashion executing different operational functionalities and communication protocols in parallel and hence are preferred for quick and efficient data retrieval and task execution over conventional single-agent systems executing the system functionalities and carrying out communication at the cost of time. The agents in MAS are considered to be autonomous entities such as software programs or robots (Wooldridge, 2009). MAS has the advantage of providing simultaneous results with cooperation, coordination and distribution of tasks to identify problems that must be resolved by each

* Corresponding author.
E-mail addresses: s5227228@bournemouth.ac.uk (N. Viswanathan), smeacham@bournemouth.ac.uk (S. Meacham), fadedoyin@bournemouth.ac.uk (F.F. Adedoyin).

https://doi.org/10.1016/j.caeai.2022.100057
Received 6 December 2021; Received in revised form 12 February 2022; Accepted 26 February 2022
Available online 11 March 2022

agent (Morais, Oliveira, & Jorge, 2012; Rodríguez Marín, Duque, & Ovalle, 2015). Adaptive learning content designed for every individual student makes a positive difference in learning management systems (LMS) where students with varying capabilities and learning mode preferences enrol to emerge as successful learners. This quality provides the urge to explore the adaptivity factor of learning management systems with the support of multiple agents which cooperatively interact with each other thus providing an adaptive LMS to learners (Vuković et al., 2021, p. 1370).

According to Hamal, Faddouli, and Harouni (2021), MAS is a potential solution for the implementation of an adaptive online education system. The research proposes to test the said hypothesis by designing and simulating a stand-alone system with intelligent agents which can be attached to an existing e-learning system that includes different pedagogical and course organising functionalities with an additional ontological agent to handle real-time decision-making situations and an online agent which will communicate the students' learning preferences entered online to the rest of the MAS. A validation tool will be developed to evaluate the performance of the system in providing adaptivity through case studies and tracking of communication between agents. The result of validation will be the recorded conversations between agents, the learning modes preferred by the student and the course content presented to the student which can validate the system for different case studies. The simulation is planned to be executed using the Java Agent Development Framework (JADE) since the platform is FIPA-compliant.

The rest of the paper is organised as follows: The following section gives a brief explanation of the existing research work on the different concepts employed in implementing adaptive multi-agent online education systems and their advantages and drawbacks. Sections 3 and 4 describe the methodology followed to arrive at the proposed system and implementation of the proposed system respectively. Section 5 provides detailed evaluation strategies with the help of case studies. The study ends with a conclusion and future work section to provide a summary of the obtained knowledge from cited papers and implementation followed by areas for future study on the subject.

2. Related work

LMS are used by most universities, organizations and institutes all around the world to support and enhance e-learning capabilities. Some of the facilities provided by LMS are discussion forums, chat rooms, video conferences, weblogs, file sharing and resource recommendation (Amane, Aissaoui, & Berrada, 2021). LMS is a strategic solution for planning, creation, delivery, management and maintenance of all learning courses or events within an education setup. A good LMS provides a dynamic environment for creating interaction between learners and instructors. They are capable of providing course content based on the learning mode of students (Jamili oskouei, 2014). Traditional LMS offer little innovation and hence to incorporate the mentioned principles, adaptation is added through new agents, say, intelligent agents, which generate questions automatically, using which recategorization is performed to decide on the educational resources (LO) to be provided for students based on their constantly changing learning preferences (Al-Omari, 2017; Elghibari, Elouahbi, & el Khoukhi, 2019).

Multiple papers have been published on adaptive multi-agent educational systems in the past decade. According to Selmi, Brahmi, and Gammoudi (2017), the challenge with adaptive MAS with regards to building a VLE is to keep track of changes in the behaviour of agents and maintain a trust relationship between them. However, maintaining such a relationship among agents is a major challenge in designing a MAS (Neto, 2017). On the other hand, authors have also stated the advantages of employing a MAS in contributing to a majority of the important functionalities in VLEs. Bednarik et al. (2005) confront that multi-agent education systems can update information on websites based on current trends which are seen as a huge advantage in terms of the intelligence of the system. They also point out that multiple agents help in managing the complexity of the educational domain and the rendering of elements of the domain with efficiency and speed when compared to other systems.

Regarding the significance of adaptivity in a Learning Management system (LMS), Li, Chang, Chu, and Tsai (2012) have proved via a survey taken from online learners that, over 93% were fully satisfied with the personalisation of the content provided to them which is a higher percentage than that of the course recommendations provided by traditional non-adaptive educational recommender systems. Giuffra and Silveria (2013) point out that Intelligent Tutoring Systems (ITS) respond to students' learning needs by closely analysing their activities and providing an adaptive learning experience to every student. They keep updating the student model with the acquired knowledge then and there, thus constantly providing interesting resources and content. Trojahn and Osorio (2004) propose an intelligent agent to be added to existing VLEs just for adaptivity. The agent automatically categorises content for user navigation using machine learning techniques which is a promising addition to a VLE. Al-Omari (2017) proposes a hybrid architecture design which is a novel approach in providing dynamic real-time adaptivity in any LMS based on learners' learning mode.

Unlike previous studies, this paper stands out by providing a detailed explanation of Simulation relation which is considered as an important criterion for the implementation of a good MAS. According to Bednarik et al. (2005), Melesko and Kurilovas (2017) and Al-Omari (2017), the Felder-Silvermann Learning style model is considered the best for providing educational content type recommendation, especially for engineering courses. The model has four classifications of learning schemes, namely visual/verbal, sensing/intuitive, active/reflective and sequential/global. Each of the course materials has the related learning modes included in the metadata and the profile of the user or learner provides the preferred learning scheme resulting from the answers to a registration questionnaire, preferably the Soloman-Felder questionnaire. The user's learning objectives are matched with resources' metadata by calculating the Suitability Index and thus the most suitable course content is displayed as per the learning objectives of the user recorded through the questionnaire. The importance of learning mode has been described by Morales-Rodríguez, Ramírez-Saldivar, Sánchez Solís, and Ramírez (2012) where the intelligent agent being added to a VLE chooses the teaching strategy based on the student's learning mode thus filtering learning objects for every student accordingly. This approach seems effective as it has been followed in most of the cited works, namely (Giuffra & Silveria, 2013; Jamili oskouei, 2014; Morales-Rodríguez et al., 2012).

FIPA (Foundation for Intelligent Physical Agents) is a non-profit organization that takes care of standardization issues about agents and MAS. It has set out stringent services that enable every enterprise to become a Node in an interconnected network of Agent cities. Every enterprise is bound to have its software implemented on FIPA compliant agent platforms thus enabling communication among agents on different platforms and hence access to required services. The essential services provided by FIPA specifications in an agent platform are the Directory Facilitator (DF), Agent Management System (AMS) and Message Transport Service (MTS) (Sandita and Popirlan, 2015).

There are many MAS toolkits available in the market including those based on Java and other web relevant technologies, say JADE, JAS, JACK, ZEUS, LEAP, ADK, AAP, Comtec, FIPA-OS, Grasshopper, XML, SMTP Active Objects, etc. JADE is an open-source java-based platform for building agent-based distributed systems under the Library Gnu Public Licence. JADE was developed by Telecomm Italia lab with the University of Parma Computer Engineering group (Bellifemine, Bergenti, Caire, & Poggi, 2005). A set of software abstractions and tools have been used by JADE for the implementation of FIPA specifications. These abstractions enable developers to create code that complies with the specifications despite hiding the actual specifications from developers. The abstractions are implemented using an object-oriented language called Java since agents share the properties of OOPS, say encapsulation, message passing, inheritance etc. and hence any OOPS language can be used to code a MAS (Sandita and Popirlan, 2015). JADE implements DF and AMS as individual agents which are present in the main container. An agent platform can have one or more containers across systems or in the same system in addition to a mandatory main container, thus making it possible to create distributed agent platforms (Kristensen, Bech, & Dyngeland, 2013). The packages of JADE provide the required basic support for MAS simulation. Agents ought to share the same semantics for effective communication. Hence the exchanged conversations must-have content written in a common language with the same ontology (Cuesta, Gómez-Rodr'iguez, & Rodr'iguez-Mart'inez, 2004).

Agent's communication occurs by sending (acting) and receiving (perceiving) messages. Message passing is the means of communication between agents. KQML (Knowledge Query and Manipulation Language) is the communication language that is used to create messages as objects and send them across to the target agent in the same or different system. Agents process the received messages using intelligence which is gained through knowledge (LOPES, Cortes, Gonçalves, & OLIVEIRA, 2018). The application proposed by Sandita and Popirlan (2015) uses collaborative agents to access the required data from remote distributed databases locally, thus analysing them and extracting the required information without the compelling need to transfer the data across the network. JADE takes care of managing the agent communication in the implementation of MAS and supports agent mobility. An issue with the paper is that more practical queries like the request for files is not implemented and is mentioned as future work (Iglesias, Moreno Novella, Ricci, Diego Rivera Pinto., & Roman, 2020).

Al-Omari (2017) proposes a MAS architecture based on Event-Condition-Action (ECA) model which is implemented as an extension to Moodle to introduce personalisation/adaptivity. It is a reactive model that responds in real-time to the environment. It can react to the environment promptly by using a simple rule-based approach. When a student accesses a course in the LMS or submits a quiz, these acts are recognised as events in the LMS and can trigger pre-defined rules. The condition component can be seen as a decision component to check whether the condition is set to TRUE. If so, the action component responds accordingly and corresponding tasks are executed. In this perspective, the completion of the Felder-Silvermann learning mode questionnaire is recognised as an event happening in the LMS that is part of the adaptation process. A detection mechanism is developed so that necessary database triggers could provide the MAS with the required data about events as files. The MAS monitors any new text files in a specific folder called the "Events Repository" to act upon it. The resulting content is presented using the Adaptive Interface Agent (AIA). However, the study could be improved by adding online agent interaction in place of database triggers to provide a more real-time decision on the learning mode preferences in a VLE. The proposed system aims to correct the said drawback by introducing an online agent for communicating web results to the MAS unit.

Holgado-Terriza, Pico-Valencia, and Garach-Hinojosa (2020) propose a system that exhibits communication between external entities (Java web application) and the distributed MAS agents to fulfil a set of goals. The mechanism makes use of the JadeGateway and Gateway-Agent classes provided by the JADE toolkit which encapsulate the messages and pass on the same between the web pages and the MAS as shown in Fig. 1.

The results of submission on a web page are passed on to a Java servlet which in turn invokes the Jade Gateway agent with the results to be passed on. Once the Jade Gateway agent is successfully invoked, the HTTP results are communicated to the MAS followed by the action of the MAS in producing the appropriate results back to the browser for user interaction. However, the paper does not provide a full-fledged case study implementation which is a drawback (Holgado-Terriza et al., 2020).

Following the discussion on different agents employed in the mentioned research papers to increase the efficiency of VLE, the importance of having intelligent agents in a web application (online education system) that communicate the HTTP results to the distributed MAS is explored for implementation (Holgado-Terriza et al., 2020). The idea of interaction between online agents and internal MAS has been incorporated in our proposed system without compromising on data security. The ECA model proposed by Al-Omari (2017) is worthy to be considered for our proposed system. Other aspects say the Felder-Silvermann learning mode detection agent, adaptive course content organising agent and a control agent handling all the other agents of the MAS have also been planned for implementation in our proposed system. The existing paper does not consider having intelligent agents on the e-learning web page, and the concept is attempted in our proposed system.

3. Methodology and system architecture

The proposed system is designed using a multi-agent approach including an online agent which will pass on the messages from a VLE to the MAS system. The proposed MAS comprises the following agents for providing an adaptive learning platform for learners based on learning mode: 1. Jade Gateway Agent (Online agent), 2. Control Agent, 3. Learning Style Detector Agent, 4. Adaptive Course organiser Agent (ACA) and 5. User Interface Agent.

The online web application agent (1) has been added as per the guidance provided in (Kelemen, 2006) and work presented in (Holgado-Terriza et al., 2020), where the importance of communication between online agents and the distributed MAS in an online education system is vividly explained with illustrations. Agents 2, 3, 4 are inspired

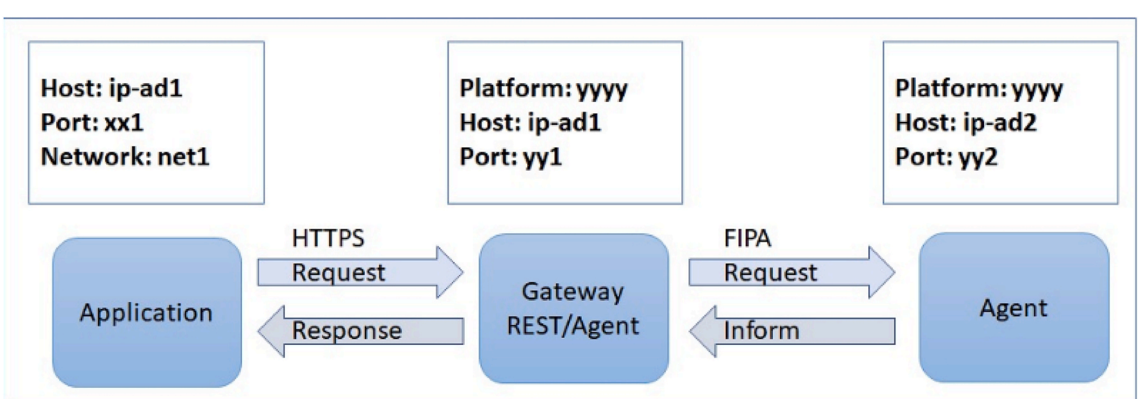

Fig. 1. Online Application - MAS communication (Holgado-Terriza et al., 2020).

by the work of Al-Omari (2017) where similar agents are implemented to provide an adaptive course structure to learners based on their learning mode preferences which looked promising for an adaptive extension to a VLE. The User Interface agent (5) is designed to render the result of the remaining agents on the browser. Finally, the ECA model approach has been incorporated from the work of Al-Omari (2017) since it reflects the real-time event-action handling in a VLE.

3.1. System design

The two main blocks of the system are the learning management system and the proposed MAS. The bridging layer is the online message passing unit which comprises an online agent which plays the key role of conveying the online submissions to the appropriate agent in the MAS. The proposed system interacts with the database for storage and retrieval of learner and course data.

As depicted in Fig. 2, the proposed system is a stand-alone unit that can be employed as an extension to an existing learning management system (LMS) to improve the adaptivity of the learning environment. The system comprises an event-condition-action unit that houses an online agent focusing on request redirection, as a result of an online event triggered by a learner, to the appropriate agent in the MAS. The critical functional unit is the MAS with 5 collaborative agents.

3.2. Database design

The database is developed using MySQL and the following Entity Relationship diagram explains the overview of the same.

As shown in Fig. 3, the Entity-Relationship (ER) diagram marks 6 entities that are related to each other: the Learner entity is related to the Questionnaire, Learning Style, LS Assessment and Course entities since every registered learner has a role to play with each of them; the Questionnaire entity is also related to Learning Style since every question has an underlying learning mode associated with it which decides the preference of the user based on the response; the LS_Assessment entity, which is the Self-Assessment questionnaire, is related to learning mode for the same reason as that of the Questionnaire entity; we can also deduce a relationship between the ContentCategory entity and the LearningStyle entity since the different categories of contents say Forums, Quizzes, Exercises etc. support particular learning modes with a support score that decides the filter and order of course contents for the learner; and similarly, the Course entity is related to the ContentCategory entity meaning that every course has one or more files for each content category to render the computed number of files for each category based on learner's preferences.

The purpose of different tables is provided in detail in Table 1. The fields marked with an underline are the identifiers (primary keys) of the respective tables and the ones marked in Italics are the fields dependent on other tables (foreign keys).

The steps taken in utilising the designed system and database towards the goal is explained in the implementation section which follows.

4. Implementation

This section shows how the components of the proposed system architecture work together to improve the adaptivity of a VLE. The technologies used in the system to achieve the results are explained in this section. The concepts of intelligent agents and web-based agent technology are discussed.

The components of the system, as shown in Fig. 4, include two main sections – the web application (VLE or LMS) and the MAS. The entire system is developed using the JAVA web application as the front end and business logic, and MySQL technology as the backend (database). In the perspective of the above figure, the web application or the VLE demo application is developed using JAVA – JSP and Servlet technology and the critical MAS is developed using the JADE MAS toolkit in JAVA technology. The data is stored and retrieved using the MySQL database. The following sections describe the different components of the system

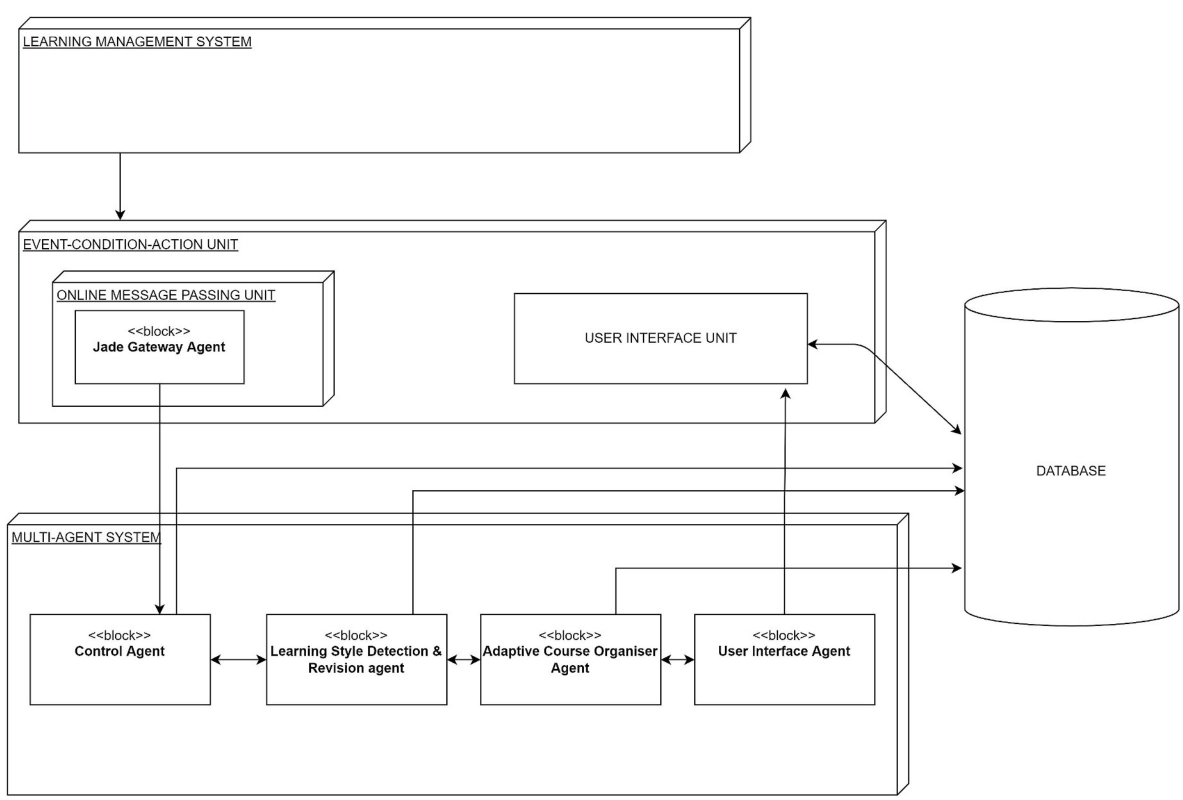

Fig. 2. Low level architecture diagram.

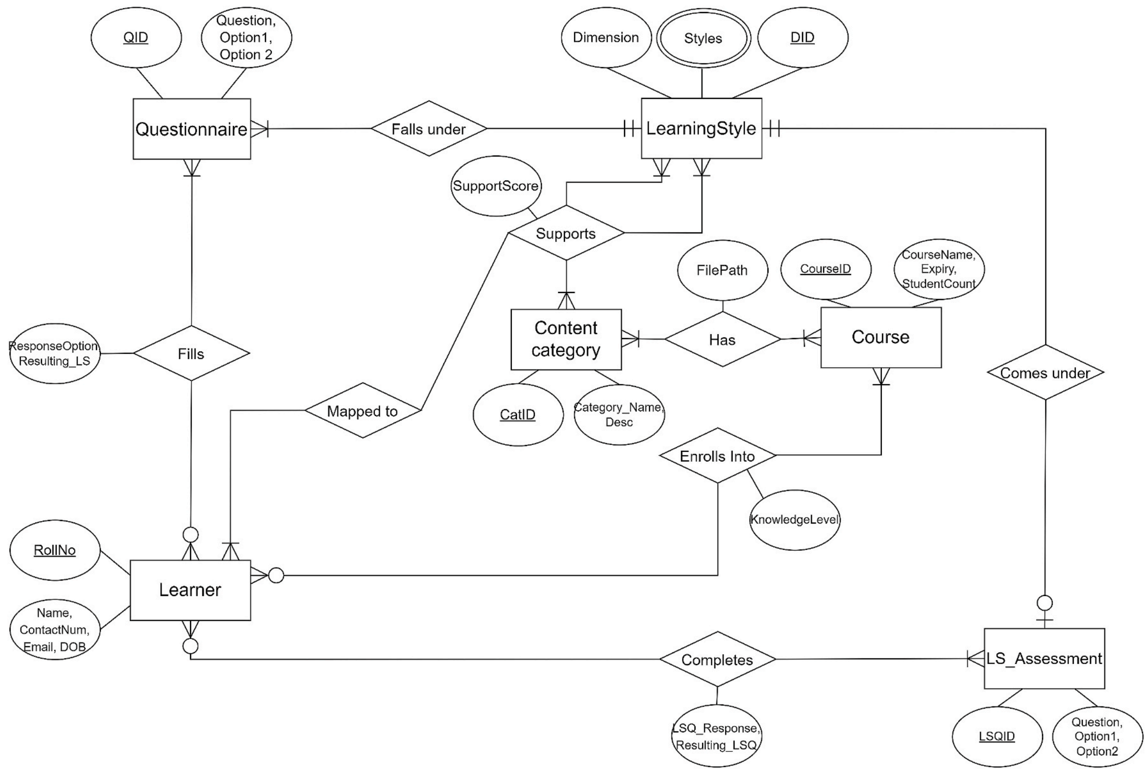

Fig. 3. Entity relationship diagram.

in detail to provide a clear view of the actual functioning of the system as a whole.

4.1. Learning modes questionnaire

The system introduces adaptivity by using the Felder-Silvermann learning style model, where it presents Soloman-Felder questionnaire with 44 questions each of which determines the values of learning mode dimensions, namely Processing (Activist/Reflective), Perception (Sensing/Intuitive), Input(Visual/Verbal) and Understanding(Sequential/Global). Table 2 gives an overview of the 8 learning mode classifications.

Each of the 44 questions of the questionnaire corresponds to one of the four LS dimensions and hence the response is mapped to the appropriate learning mode.

Once a learner registers into the system, the LS questionnaire is presented to them followed by 3 more pages of the questionnaire with 44 questions. On submission of responses, the online agent and the rest of the agents in the MAS cooperate to fetch the most appropriate course contents for the learner and the algorithm employed by each of the agents is explained in the upcoming sections.

4.2. Self-assessment questionnaire

The self-assessment questionnaire is made available to registered learners to be completed anytime during the course for their LS preference changes. It is a simple questionnaire with 4 questions, each of which is associated with an LS dimension similar to the Solomon-Felder questionnaire. The questionnaire is purposefully made simple to avoid skipping questions by learners.

Based on the response for each dimension(question), the appropriate LS preference changes are recorded in the database. In both the above questionnaires, any unanswered question is mapped to “Unknown” as the value in the database. The figure also shows evidence for the same in the response(R) section.

4.3. Jade Gateway agent (online agent)

Since the paper deals with online education systems, it is worthwhile to include a web agent as part of the MAS to capture online events and pass them on to the internal agents. JADE MAS toolkit provides GatewayAgent and JadeGateway classes to build a web agent which will capture events in a JSP page and pass the results on to the first internal agent of the MAS (Holgado-Terriza et al., 2020). Once the initial LS questionnaire is completed by a learner, the submission event is detected by the corresponding servlet which in turn passes on the selected options (results) to the JADE Gateway Agent by creating a Blackboard be an object which is the message channel between online agent and servlet, and executing the JadeGateway.execute() method (Kelemen, 2006). The diagrammatic representation of the message passing action is provided in the VLE section of Fig. 5.

The communication of the web application with the web agent forms the Event-Condition-Action unit where the submission event is detected and passed on to the Jade Gateway Agent if a valid answer is marked for at least one of the questions. The Jade Gateway agent in turn passes on the 44 responses through the JadeGateway object to the Control Agent, which is part of the critical MAS unit explained in the following section.

4.4. Multi-Agent System

Multi-Agent system, which is a highly researched area in the Artificial Intelligence platform, is employed in our proposed system with a view of increasing the efficiency of an LMS. Our system houses collaborative software modules as agents in the MAS unit which process and pass on the results at every stage to the subsequent agents. The communication among agents happens through ACLMessage objects which encapsulate the message or the content object with the sender and receiver address and the message performative being sent, say INFORM, REQUEST, CONFIRM etc (Scutelnicu, Lin, Liu, Graf, & McGreal, 2007). In our proposed system, the set of 44 responses which are passed on to the Control Agent of the MAS unit is processed by 4 agents apart from

Table 1
Purpose of database tables.

S. No.	Table Name	Fields (Columns)	Purpose
1	std_questionnaire	QID Option1 Option2 *DID*	The master table houses the Soloman-Felder questionnaire with 44 questions which determine the learning style preferences of the learner. Each question has 2 options to choose from.
2	std_lsdimensions	DID Dimension	The master table holds the 4 broad learning style dimensions whose values are determined for every learner.
3	std_dim_styles	Style *DID*	The master table holds the learning styles available for each of the 4 dimensions mentioned in the previous table.
4	std_ls_assessment	LSQID Question Option1 Option2 *DID*	The master table houses a learning style revision questionnaire which revises the learning style preferences in the middle of the course based on responses to 4 questions corresponding to the 4 learning style dimensions. Each question has 3 options to choose from.
5	std_content_category	CatID Category_Name Cat_Desc	The master table holds the 8 available categories, say Quiz, Forum, Exercises, Main content etc. which are hidden or displayed to the learner based on their computed learning style preferences.
6	std_ls_cat_support	*DID* *CatID* SupportScore	The master table provides the link between a category and the supported learning style with a learning score. For example, learning style "Activist" supports Forum, Exercises and Quiz categories with a score of "1″ whereas the style supports Examples with a score of "-1". This scoring concept is introduced since different learning styles may have contradicting support for categories, say if a learner is of Activist and Sensing type, then one has a score of 1 and the other has −1 which means that moderate examples need to be rendered for the learner.
7	std_course	CourseID CourseName CourseExpiry StudentCount PassPercentage	The master table holds the available courses in the system to enable learners to enrol themselves.
8	std_course_cat	*CourseID* *CatID* FilePath	The master table holds the path of all the files to be rendered for the available courses under every category. For example, the quiz file paths for AI course, exercise file paths for Data Analytics course and so on.
9	learner	RollNo LName Fname DOB EmailID ContactNumber	The table records the details of every learner who have registered into the system.
10	learner_course	*RollNo* *CourseID* KnowledgeLevel	The table records the course (s) in which every learner is registered.
11	learner_quest	*RollNo* *QID* ResponseOption Resulting_LS	The table records the responses of learning style questionnaire for every registering learner and also the resulting learning style preference based on each response.
12	learner_assessment	*RollNo* *LSQID* LSQ_Response Resulting_LSQ	The table records the responses of self-assessment (learning style revision) questionnaire for every learner and also the resulting learning style preference based on each response.
13	learner_ls	*RollNo* *DID*	The table records the calculated learning styles for initial as well as revised LS under each of the 4 LS dimensions for every learner.
14	learner_course_contents	*RollNo* *CourseID* *CatID* FilePath	The table records the final determined course contents to be delivered to every learner on the browser in the determined order of priority.

the web agent which is listed as Control Agent; Learning Style Detector/Revision Agent; Adaptive Course Organiser Agent; and User interface Agent. The agents send and acknowledge messages by collaborating thus carrying out their tasks and passing on the intermediate results to the immediate next agent. The overall activity diagram of the MAS is present in Fig. 5 for a broad overview before getting into details in the following sections.

4.4.1. Jade platform

JADE MAS toolkit has been chosen to build the MAS unit since it is a widely used platform for multi-agent development and simulation. MAS comprises one or more agent platforms and each platform has one or more agent containers. Agent containers are the holders of one or more agents and every MAS has a mandatory main container which is the launch container while executing the system. The main container has very important responsibilities, namely managing the container table (CT) which houses the references and transport addresses of other containers, managing the Global Agent descriptor table (GADT) which comprises of state and location information of all agents in the platform and also managing two special agents. The special agents are Agent Management System (AMS) which is responsible for naming and creating/killing agents and the Directory Facilitator (DF) which lists the services provided by other agents for use by agents to achieve their goals (Scutelnicu et al., 2007). The GUI is illustrated as part of the supplementary materials.

The agents "ams", "df" and "rma" are the default agents of the main container which manage the registration and life cycle of the agents. The figure also presents all the system-specific agents of the MAS unit as depicted in Fig. 6. The functionality of each of them is explained in the following section.

4.4.2. Description of agents

The MAS unit of the proposed system houses four functional agents which form the heart of the study. The fifth agent which is the web agent has been explained in the earlier section and so the same will not be revisited here. The class diagram of the entire system is furnished in Fig. 6.

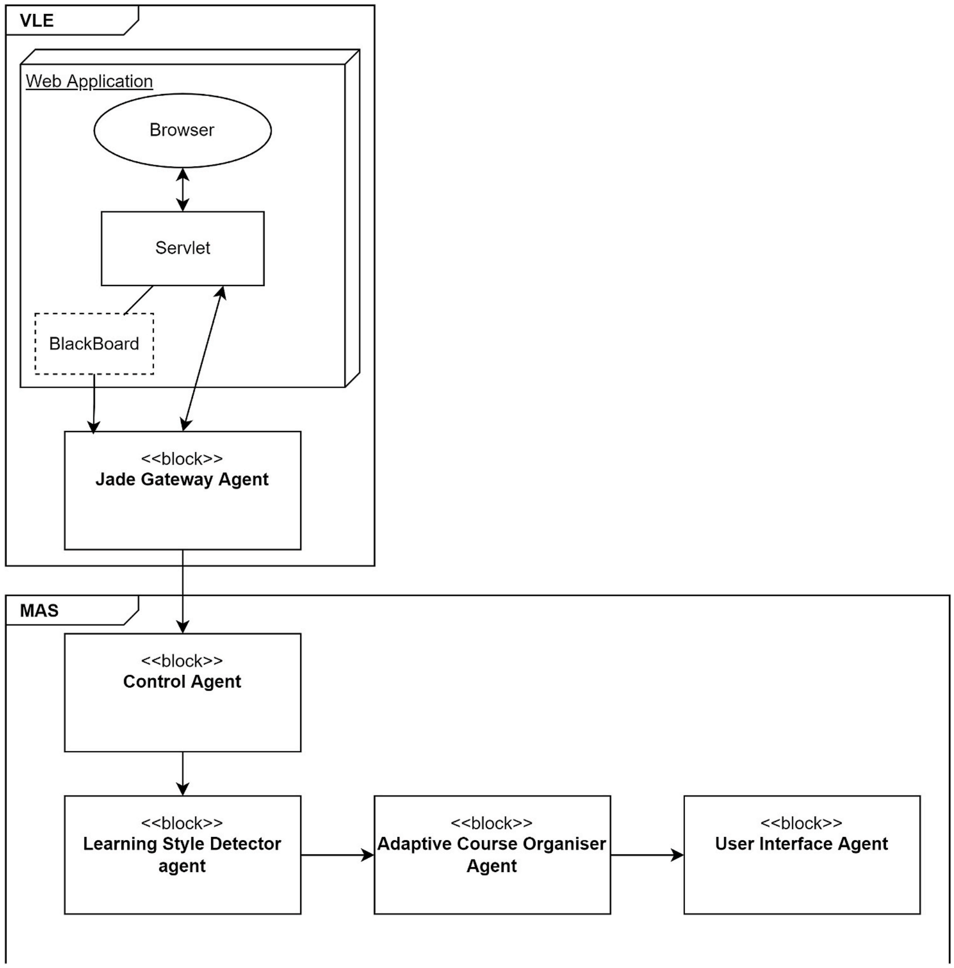

Fig. 4. System components.

Table 2
Learning Modes with description.

LS Dimension	Learning Style	Description
Processing	Activist	Learners who learn best by brainstorming, having group discussions and activities.
	Reflective	Learners who think through any topic in order to excel in the same.
Perception	Sensing	Leaners who trust facts and proven concepts.
	Intuitive	Learners who are innovative and novel in their thinking with abstract ideas.
Input	Visual	Learners who prefer to learn from visual content like videos and figures.
	Verbal	Learners who prefer to learn from verbal content.
Understanding	Sequential	Learners who follow a procedural learning habit in order to excel in a topic.
	Global	Learners who understand a topic better by reading its overview/summary than being procedural.

4.4.2.1. Control agent (CA). As the name suggests, the control agent (CA) is the initial agent of the MAS unit which acts as the entry point and passes on the message from the online agent (web agent) to the rest of the agents appropriately. On submission of the LS questionnaire during learner registration, the web agent detects the submission event and passes on the responses to the control agent as explained earlier. The control agent extracts the parameters from the ACL message and derives the resulting learning mode for each of the responses based on the dimension associated with the question. The derived learning mode results are passed on to the Learning Style detection and revision Agent.

Control Agent also plays a similar entry point role during learning mode revision of learners when a self-assessment questionnaire is submitted at any point in time. The CA converts the questionnaire responses into the corresponding learning modes and sends them over to the LDA agent.

4.4.2.2. Learning style detection/revision agent (LDA). The learning style detection/revision agent, which is named “LDA” in the system receives the 44 learning mode results from the control agent and sends back an acknowledgement confirming reception. The agent then derives the final learning mode values for the four LS dimensions based on the count of each unique learning mode result in the input message. The final learning mode preferences of the learner are also recorded in the database. The LDA agent passes on the final learning mode preferences of the learner to the ACA agent for further processing.

The LDA agent actively participates in the LS revision process in the same way. Once the agent receives the learning mode results from the CA agent, it directly updates the learning mode preference of the learner, since the four revised responses provide the exact values for LS without requiring much processing during LS revision. The results are passed on to ACA agent. The updated learning mode preference of the learner is recorded in the database.

4.4.2.3. Adaptive course organiser agent (ACA). The role of the ACA starts when the final learning mode preferences are delivered to the agent by the LDA. Every learning mode supports few course content categories, say Forums, Quizzes, Exercises, Examples etc. The database houses a master table that stores the relationship between course

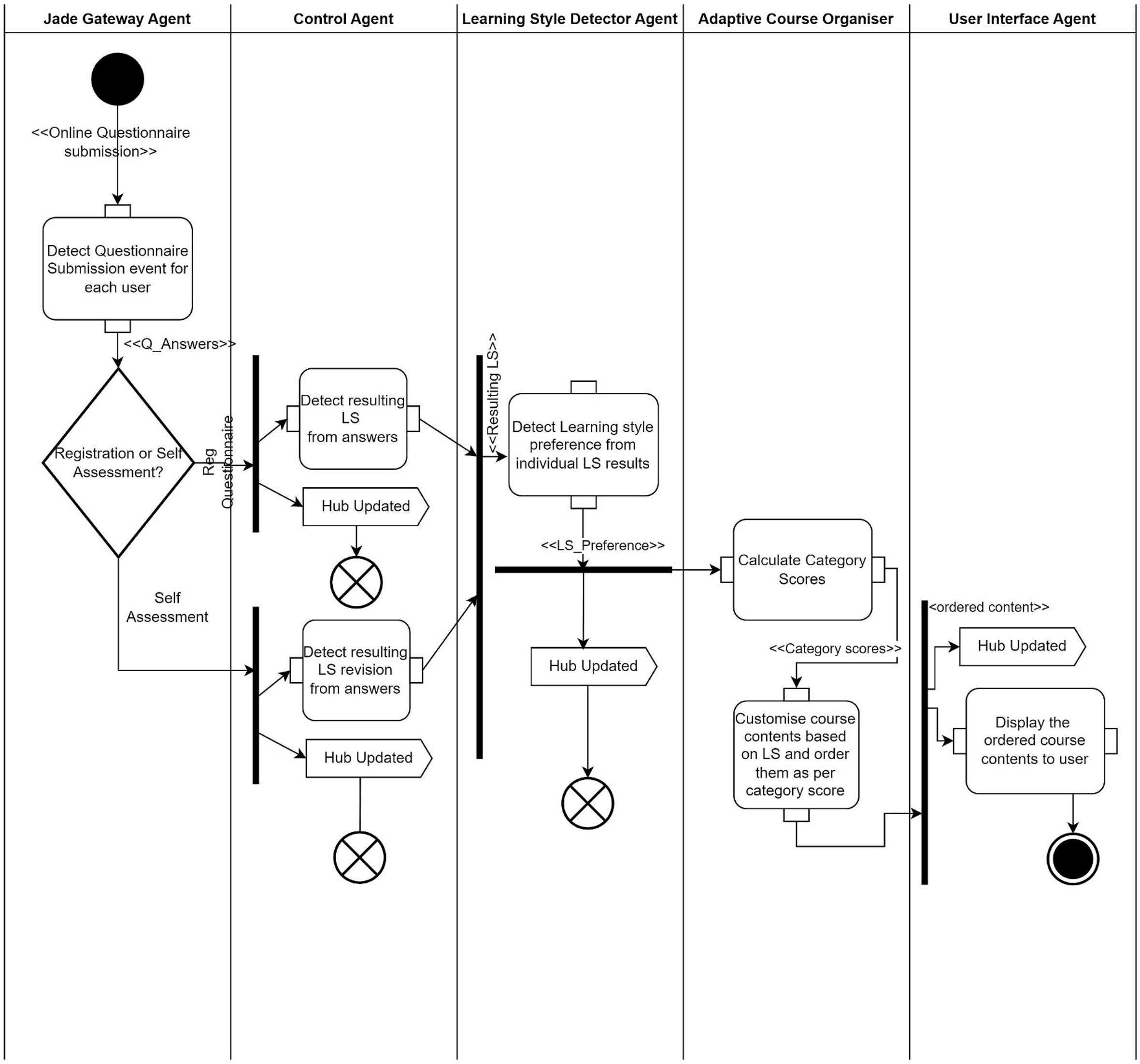

Fig. 5. Activity diagram of the MAS.

category and the supported learning mode preferences as shown in Table 3. For example, a learner with an Activist learning mode would prefer Forums, Exercises and quizzes but would not prefer examples as part of the course content. Hence, we have assigned a support score for all learning modes against every course category in a master table. ACA refers to the said master table to calculate the sum of support scores for every category based on the input learning mode preferences. The support score summation mechanism is followed in the study to justify the category support contradictions among learning modes under different learning mode dimensions. For example, if a learner prefers Activist and Sensing learning modes, the support score for the category Examples is −1 and 1 respectively, meaning that one supports and the other does not support Examples. In such cases a moderate number of examples will be provided to the learner as the sum of scores is not negative and it is "0" instead. Thus, the ACA calculates the total support score for each category under the registered course of the learner. Finally, the contents corresponding to the course under each identified category are prioritised based on the support score and passed on to the UI agent for rendering on the browser.

The agent shows the same behaviour for the LS revision process since the input to the agent is in the same pattern as that of the initial LS detection.

4.4.2.4. User Interface Agent (Ui). The UI agent simply receives the ordered list of course contents from the ACA agent and records them in the database for rendering on the browser on demand. Only the file paths are rendered here in place of actual files because the aim of the study is the only simulation of MAS and the actual development of the product is considered as future work. The final course content list rendered to the learner reflects his/her learning mode priorities for the registered course and the contents are ordered based on the degree of interest towards the content categories, thus motivating the learner to utilise the resources to the maximum and score better every time.

An overall picture of the sequence of message transfers and functionalities of the MAS is provided in the sequence diagram furnished in Fig. 7. The process starts with a learner registering in the portal with his Roll Number and course details, followed by the corresponding Java Servlet taking forward the questionnaire results to the Jade Gateway Agent (online agent). The Jade Gateway Agent communicates the responses to the Control Agent in the internal MAS with the help of JadeGateway class provided by the JADE MAS toolkit. The process continues with interim results being transferred among the agents in the MAS to derive the customised course content results for the learner. The process ends with rendering the customised course files concerning the type of content, the quantity and order of content on the web browser based on their interested learning mode for the registered course.

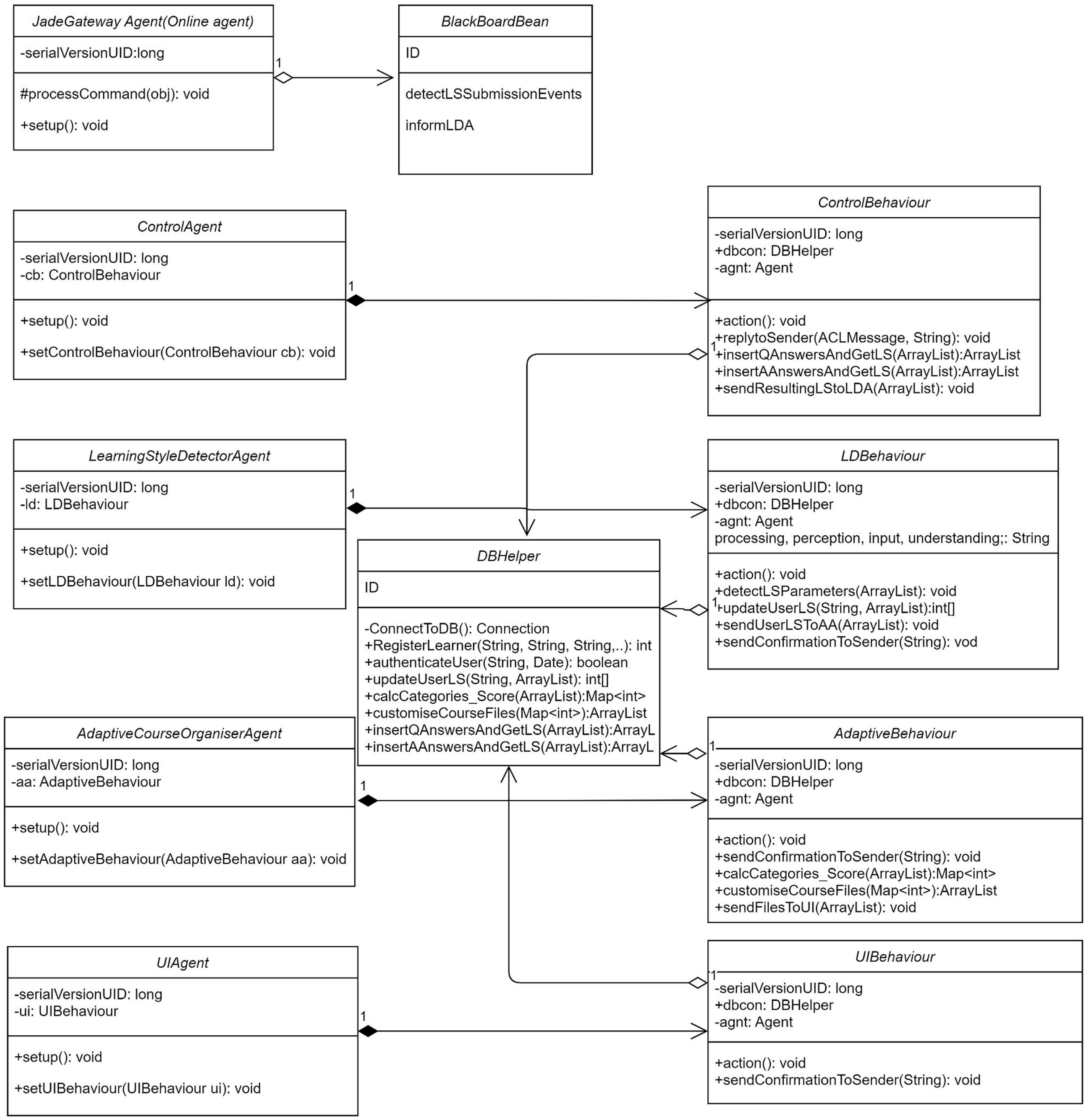

Fig. 6. MAS - Class diagram.

5. Evaluation and results

The MAS is evaluated using the graphical tools provided by the JADE MAS toolkit. We will use the Sniffer agent tool to sniff the conversations of all the agents to evaluate the agent communication during the process of deriving the adaptive course contents for learners. In addition to sniffing the agents, we will also record the actual messages which are transferred between agents and validate against the recorded learning mode preferences. We will consider few critical scenarios as explained in the following case studies to bring out the processing steps and message transfers of the proposed adaptive MAS extension so that the system can be validated against the learning mode preferences of the learner.

1. A learner with Activist, Sensing, Visual and Sequential learning mode preferences turns into a verbal learner
2. A learner with Activist, Intuitive, Visual and Global learning mode preferences answering without any difference in the self-assessment questionnaire

The case studies are studied in detail with relevant pieces of evidence in the following section.

5.1. Case studies

5.1.1. Case study #1

In this case study, we will see the sequence of operations and message transfers happening for a learner who registers with **Activist, Sensing, Visual and Sequential** learning mode preferences initially and turns into a **Verbal** learner later on. We will respond to questionnaires such that the resulting learning modes are the ones in the case

Table 3
Learning mode- category support.

LS/Category	Forum	Examples	Exercise	Quiz	Visual	Verbal	VisualVerbal	Summary
Activist	1	-1	1	1				
Reflective	-1	1	-1	-1				
Pr-Balanced	0	0	0	0				
Sensing		1	1	1				
Intuitive		-1	-1	1				
Pe-Balanced		0	0	0				
Visual					1		1	
Verbal						1		
Ip-Balanced					0	0	0	
Sequential								-1
Global								1
Und-Balanced								0

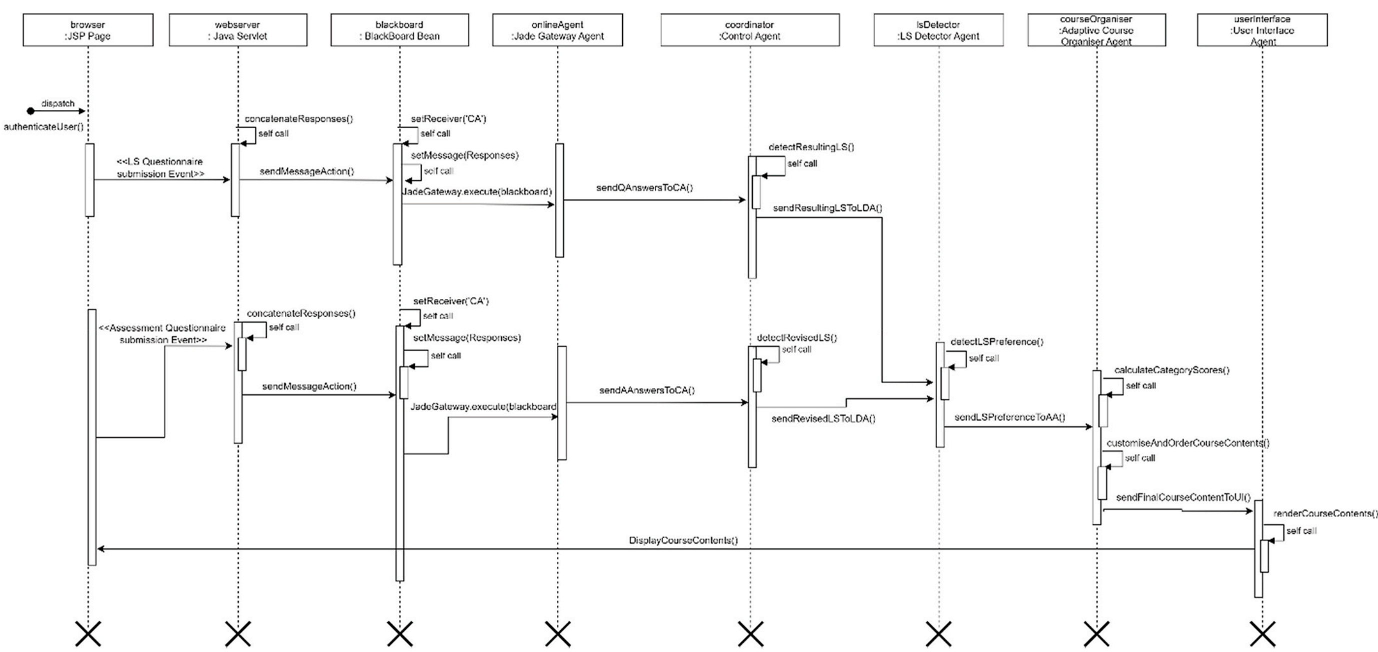

Fig. 7. Sequence Diagram of the system.

study requirement. Initially, the registration questionnaire is answered by a new learner.

The learner with Roll number #45 has answered the registration questionnaire as shown in Table 4. We can see the agent interactions where agents CA, LDA, ACA and UI derive and pass on the intermediate results of the system thus achieving the goal of displaying the appropriate course contents on the browser. Since the learner is an activist, he is interested in Quizzes, Exercises and Forums but not in examples, however being a sensing learner, examples are preferred which is contradictory and hence moderate number of examples are provided (1 file here) and three quizzes and three exercise files are provided. Also, being a Visual learner, a PowerPoint and video file deliver the main course content and being a Sequential learner, a summary file is not preferred and hence not rendered. Considering the learner's choices, the type of files and number of files are customised and displayed in the determined order of priority based on the support score for each category.

Table 4
CS1-Recorded LS preferences Vs Displayed Course Contents.

RollNo	Dimension	Learning Mode
45	Processing	Activist
	Perception	Sensing
	Input	Visual
	Understanding	Sequential
Category	**Files displayed**	**Order**
Quiz	3 quizzes	1
Exercises	3 exercises	2
VisualVerbal	1 Powerpoint file	3
Visual	1 video file	4
Forum	Access to forum granted	5
Examples	1 example file	6

The displayed course contents agree with the learning mode preferences recorded for the learner in the database as shown in Table 4.

On submission of self-assessment questionnaire at a later point, where the **Input** dimension changes from **Visual** to **Verbal**, the following interactions happen in the system thus revising the course contents for the learner.

Due to the change of Input dimension from Visual to Verbal, the resulting change in course contents is that in place of PowerPoint and video files, a verbal file in PDF format is rendered as the main course content. Table 5 shows that the revised learning mode is reflected in the course contents list rendered to the learner.

The sniffer agent is activated to sniff all the agents in the MAS so that we can view the evidence of message transfers. Fig. 8 furnished below shows the sniffer output for the entire process of registration and learning mode revision. We can see from the evidence that the agent "Other" (online agent) sends the questionnaire responses as a message of type REQUEST to Control Agent(ca) which sends back a message of type INFORM to the online agent as acknowledgement. Following this, the CA sends the resulting learning mode for each response as INFORM message to LDA which acknowledges the CA through a CONFIRM message. Then the LDA sends the final learning mode preferences of the learner as an INFORM message to ACA which acknowledges the LDA with a CONFIRM message. Finally, the ACA sends the customised course contents for the learner as an INFORM message to the UI agent which acknowledges ACA through a CONFIRM message. The contents are then rendered on the web browser for the learner.

5.1.2. Case study #2

In this case study, we will see the sequence of operations and message transfers happening for a learner who registers with **Activist, Intuitive, Visual and Global** learning mode preferences and **remain with the same preferences** even after answering the self-assessment questionnaire later on. We will respond to questionnaires such that the resulting learning modes are the ones in the case study requirement. Initially, the registration questionnaire is answered by a new learner.

The learner with Roll number #35 has answered the registration questionnaire in such a way that the resulting learning mode preference is recorded as **Activist, Intuitive, Visual and Global**. We can see the agent interactions where agents CA, LDA, ACA and UI derive and pass on the intermediate results of the system thus achieving the goal of displaying the appropriate course contents on the browser. Since the learner is activist in nature, he is not interested in Examples, but very much interested in Quizzes, exercises and interactive forums, however being an intuitive learner, he is not interested in exercises which is contradictory and hence a moderate number of exercise files are provided (1 file is provided). Since the quiz is supported by both the learning modes, three quiz files are provided, and the forum is also rendered to the learner. Also, being a Visual learner, PowerPoint and video files are delivered as the main course content and being a Global learner, a summary file is preferred and hence rendered to the user. Thus, considering the learner's choices, the type of files and the number of files are customised and displayed in the determined order of priority based on the support score for each category.

Table 5
CS1-Revised LS preferences Vs Displayed Course Contents.

RollNo	Dimension	Learning Mode
45	Processing	Activist
	Perception	Sensing
	Input	Verbal
	Understanding	Sequential
Category	**Files displayed**	**Order**
Quiz	3 quizzes	1
Exercises	3 exercises	2
Verbal	1 document	3
Forum	Access to forum granted	4
Examples	1 example file	5

The displayed course contents agree with the learning mode preferences recorded for the learner in the database as shown in Table 6.

On submission of the self-assessment questionnaire at a later point, where the learning mode preferences do not change, the resulting course contents are not altered as a result of the LS revision process. The following interactions happen in the system thus processing course content revision for the learner. Table 7 shows that the revised learning mode is reflected in the course contents list rendered to the learner and there is no difference in the learning modes in this case study.

The sniffer agent is activated to sniff all the agents in the MAS so that we can view the evidence of message transfers. Fig. 9 furnished below shows the sniffer output for the entire process of registration and learning mode revision. We can see from the evidence that the agent "Other" (online agent) sends the questionnaire responses as a message of type REQUEST to Control Agent(ca) which sends back a message of type INFORM to the online agent as acknowledgement. Following this, the CA sends the resulting learning mode for each response as INFORM message to LDA which acknowledges the CA through a CONFIRM message. Then the LDA sends the final learning mode preferences of the learner as an INFORM message to ACA which acknowledges the LDA with a CONFIRM message. Finally, the ACA sends the customised course contents for the learner as an INFORM message to the UI agent which acknowledges ACA through a CONFIRM message. The contents are then rendered on the web browser for the learner.

An overall case study summary is provided in Table 8 for reference.

5.2. Results of evaluation and significance of the proposed system

As a result of the evaluation of the discussed case studies with the evidence of message transfers and sniffer outputs, the proposed system proves to work as per the requirements of an adaptive education providing system. The proposed system can be attached to any VLE or LMS to introduce course content customisation adapted to the learners' learning mode preferences which keep changing with time. Since the technology used in JAVA is platform-independent, the compiled code can be used in any platform which adds to the merits of the proposed system. The system can be considered unique in the way it handles learning mode detection and revision as there is no published work to the best of my knowledge which provides dynamic real-time adaptivity with hybrid architecture involving consolidated features of initial learning mode detection and learning mode revision capabilities with the help of online agents interacting with the internal MAS. The introduction of web agents which help in communicating the online submission results to the MAS is a key addition to any Online education provider. The feature of handling unanswered questions in the provided questionnaires is an added advantage due to its flexibility. The initial feedback was provided by an educator for the simulated study which reads as follows: "The demonstrated system is very promising. The only problem might be the number of resources that will be required by the educator. However, this solution is optimized according to a range of learning modes so the number is less than would normally be required and the target is a purely online adaptive course which makes the requirement for an increased number of resources inevitable." Feedback from students and educators have also been received through a feedback questionnaire and the responses are recorded in the below furnished link.

https://docs.google.com/spreadsheets/d/1V6WzpefrZHFfpEbclF7rTD-ChMR5VXlV-mQ-7kYfyKo/edit?usp=sharing

The feedback are also listed here with their implications to the system as per our understanding.

FEEDBACK	IMPLICATION ON THE SYSTEM
I have tried the web page given to me and I find it very interesting to decide my own learning style based on common non-technical questions. Though filling	From the feedback we understand that the system functions flawless in determining the preferred learning mode of the user and in customising the

(continued on next page)

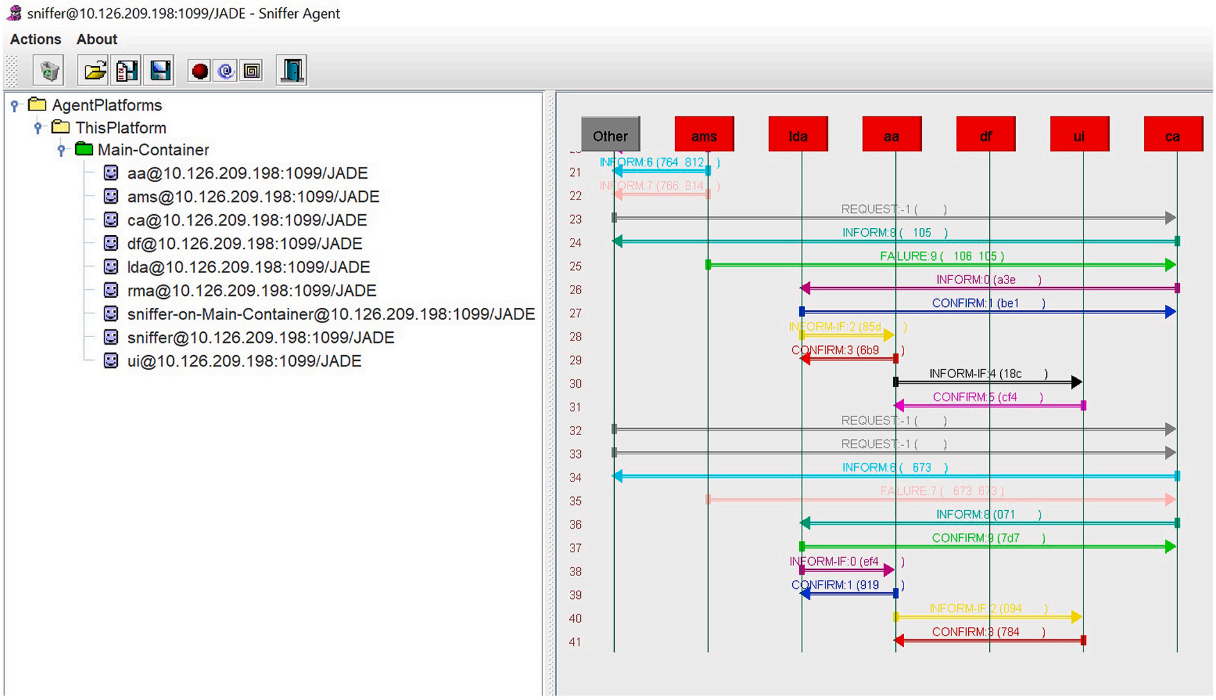

Fig. 8. CS1-Sniffer Agent showing evidence of agent conversation.

Table 6
CS3-Recorded LS preferences Vs Displayed Course Contents.

RollNo	Dimension	Learning Mode
35	Processing	Activist
	Perception	Intuitive
	Input	Visual
	Understanding	Global
Category	**Files displayed**	**Order**
Quiz	3 quizzes	1
VisualVerbal	1 Powerpoint file	2
Visual	1 video file	3
Summary	1 summary document	4
Forum	Access to forum granted	5
Exercises	1 exercise file	6

Table 7
CS3-Revised LS preferences Vs Displayed Course Contents.

RollNo	Dimension	Learning Mode
35	Processing	Activist
	Perception	Intuitive
	Input	Visual
	Understanding	Global
Category	**Files displayed**	**Order**
Quiz	3 quizzes	1
VisualVerbal	1 Powerpoint file	2
Visual	1 video file	3
Summary	1 summary document	4
Forum	Access to forum granted	5
Exercises	1 exercise file	6

(*continued*)

FEEDBACK	IMPLICATION ON THE SYSTEM
up the questionnaire consumes time, the result is worth the wait. The course content is displayed exactly in the preferred learning mode. I'm a visual person and I have got my course content as videos and presentations.	course contents displayed on the web page accordingly. However we also understand the pain in answering 44 initial questions for the system to function efficiently. In total, the user is happy with the resulting course content which is satisfactory.
I feel that the MAS system would cater to all learning capabilities because the questionnaires presented to users cover all the known learning modes and is relevant to all time periods including the future. Hence I would say that the educational MAS system would work effectively on implementation of the current design.	The feedback is positive with respect to the learning modes covered in the questionnaires. The feedback also confirms that the system is adaptive with the advancing time with respect to customisation of course contents.
My answer to the previous question is "Maybe" because the different learning modes which are required by the learners all over the world is way too higher than what is humanly understandable. However, as per my level of understanding, the system covers the needs of all kinds of learners with accuracy and speed up to the mark. Since the learning modes are always relevant to any time period, be it past, present, or future, the system will succeed on implementation of the design I see through the web page and MAS simulator.	The feedback is critical as it explains the need to analyse the other probable learning styles that may affect the understanding of students. As per our understanding and as per the existing papers on Felder-Sivermann learning style model, all learning style classifications have been covered in the paper. The user seems to be happy with the conversation exchanges endorsed by the MAS simulator and the working of the mockup web page which defines the working of the to-be-implemented add-on.
To sum up my feedback based on the above questions, I find the huge questionnaire which is presented initially a pain to complete, however on completion when I view the course content listed to me at once, I'm amazed at the content type delivered with good speed which is very much my go-to learning mode. When I wanted to switch from Visual to verbal content and answered the self-assessment questionnaire, I could see the required change in the course content list. I also tried to remain in the same learning	The feedback mentions the initial questionnaire as a pain as it is time consuming and we take it as an improvement to take up for future developments. The feedback also complements the resulting course content and the speed of delivery of results. This certifies the speed and accuracy of the MAS in working with collaborative effort. We understand that the MAS has functioned as expected when the initial choice of learning style is changed in self-assessment questionnaire.

(continued on next column)
(continued on next page)

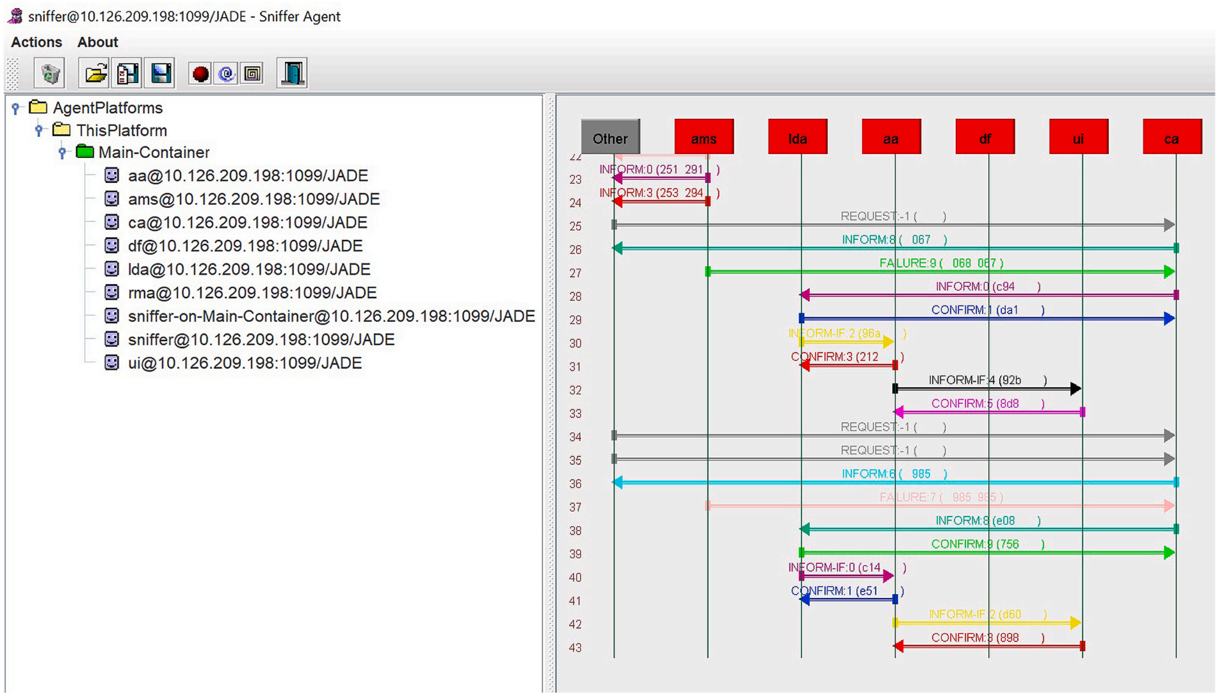

Fig. 9. CS3-Sniffer Agent showing evidence of agent conversation.

Table 8
Case study summary.

CS#	Initial LS	Initial Course Content	Updated LS	Updated Course Content
1	Activist, Sensing, Visual and Sequential	3 Quizzes, 3 Exercises, Video content, Forum, 1 Example	Activist, Sensing, Verbal and Sequential	3 Quizzes, 3 Exercises, text content in PDF, Forum, 1 Example
2	Activist, Intuitive, Visual and Global	3 Quizzes, Video content, Summary, Forum, 1 Exercise	Activist, Intuitive, Visual and Global	3 Quizzes, Video content, Summary, Forum, 1 Exercise

(*continued*)

FEEDBACK	IMPLICATION ON THE SYSTEM
mode by answering the appropriate questions in self-assessment questionnaire and found that the course content did not change as expected. The system is consistent with regards to the determination of the desired learning mode of course content irrespective of region and time.	
The quality of the system is good with regard to the decision making based on questionnaire responses. The different components of the MAS system seem to function efficiently starting from the online agent which delivers the web page answers to the internal MAS until the final agent of the MAS system which is the UI agent, based on the output of the MAS simulator and the mock up web page. The questionnaires and the logic behind the learning mode detection seem to cater to the learners of all known kinds and times. Hence the system looks promising taking all the above points in mind.	The feedback complements the use of questionnaires to arrive upon decisions. The functioning of different agents of the MAS has been complemented based on the output of the Sniffer agent (Simulator) and the corresponding course content listing on the mock up web page. The feedback also endorses the adaptivity of the system and the ability of the system to satisfy all types of learners.

(*continued on next column*)

(*continued*)

FEEDBACK	IMPLICATION ON THE SYSTEM
The MAS system can also include supporting people with disabilities in a future design	This feedback is critical since it points out the areas of improvement in the system. We will take it up as one of the items to address in our future developments.
I found the MAS system extremely effective and adaptive to learners' needs.	This is a positive feedback for the working of the system. The online agent combined with the internal MAS have been able to deliver a convincing result for most of the users who have tried their hands on the mock up web page and witnessed the MAS simulator's output.

Thus, the research provides a set of theoretical implications and in practice, it also makes use of various agents including a web agent which plays an integral part in the provision of adaptive education: message passer, controller, learning mode reviser, adaptive course organiser and course renderer. The key contribution of the system is the integration of educational technology as a teaching method and support tool, which would guide the students in their preferred pathway to achieve their goals through the MAS and VLE.

. This research carries considerable pedagogical implications for the design of an adaptive add-on for LMS and necessitates a review of the research to determine the characteristics and skills of the emerging online learner. Determining the characteristics and course preferences of the online learner may not guarantee the highest score of the learner, however it would help the learner improve his/her learning ability and understanding on the subject toa great extent as the mode of content presentation is of their interest.

6. Conclusion and future work

The paper walks through the concepts of agent technology and the use of the MAS in the field of education. We have discussed the importance of adaptivity in online educational systems which contributes to the educational growth of learners by bringing up content based on learners' interests and learning mode preferences which keep changing from time to time. In the subsequent sections, we discussed the related contribution of researchers in the implementation of MAS technology and the inclusion of online agents. The paper goes on with the description of system design and implementation. Our proposed study addresses the online education problem by simulating a MAS that can be attached to an existing LMS to provide meaningful and adaptive course content recommendations based on the learners' interests without compromising on the speed and accuracy of recommendation. The system is then evaluated to prove effective functioning and demonstrated with a detailed discussion of results.

As stated in the discussion, we see few gaps in the cited papers, where there is, say, lack of interactivity, adaptivity, debugging facilities, transparency during display of results and data pre-processing and our system aims to overcome these issues. The proposed system enables the interaction between LMS and MAS by introducing an online agent which helps in communicating the website results to the internal MAS unit. The system improves adaptivity by facilitating learning mode revision through a self-assessment questionnaire which can be answered if a user is unhappy about the current course contents. The system makes use of the sniffer tool provided by the JADE MAS toolkit to monitor and debug the behaviour of every agent. The system can be made transparent in terms of visibility of learning mode preferences of learners to teachers by making the database accessible via the configuration page of the LMS where teachers can alter the visibility of learners' results. Data pre-processing issue is overcome in the proposed system by handling the unanswered responses with a code in the database in place of a NULL value, thus avoiding NULL referencing issues in the system. Thus, the system still provides meaningful customisation of course contents despite submitting incomplete questionnaires, since the learner can alter his learning mode preferences anytime with the self-assessment questionnaire.

The future work of the study will be the development of the proposed system with the consideration of the knowledge level of the learners on the course, in addition to the learning mode preferences, to display course contents of appropriate difficulty level. Also, the learning mode preferences can be detected with the help of data mining techniques from different websites actively used by learners, rather than making them fill out a time-consuming questionnaire. Another addition could be a recommendation of useful website links and e-book links relevant to the course and subject in addition to the existing course content uploaded by teachers. These additions would make the system complete and more self-sufficient to make a valuable addition to an LMS.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Al-Omari, M. (2017). *An agent-based architecture to support adaptivity in virtual learning environments based on learners' learning styles*.

Amane, M., Aissaoui, K., & Berrada, M. (2021). A multi-agent and content-based course recommender system for University E-learning platforms. *Lecture Notes in Networks and Systems, 211*, 663–672. https://doi.org/10.1007/978-3-030-73882-2_60. LNNS.

Bednarik, R., Joy, M., Moreno, A., Myller, N., Sun, S., & Sutinen, E. (2005). Multi-agent educational system for program visualization. In *Proceedings - International conference on computational intelligence for modelling, control and automation, CIMCA 2005 and International conference on intelligent agents, web technologies and internet* (Vol. 2, pp. 283–288). https://doi.org/10.1109/cimca.2005.1631482

Bellifemine, F., Bergenti, F., Caire, G., & Poggi, A. (2005). *Jade — A Java Agent Development Framework*. https://doi.org/10.1007/0-387-26350-0_5

Chen, X., Xie, H., Zou, D., & Hwang, G.-J. (2020a). Application and theory gaps during the rise of artificial intelligence in education. *Computers and Education: Artificial Intelligence, 1*, Article 100002. https://doi.org/10.1016/J.CAEAI.2020.100002

Chen, X., Zou, D., Cheng, G., & Xie, H. (2020b). Detecting latent topics and trends in educational technologies over four decades using structural topic modeling: A retrospective of all volumes of computers & education. *Computers & Education, 151*, Article 103855. https://doi.org/10.1016/J.COMPEDU.2020.103855

Cuesta, P., Gómez-Rodr'iguez, A., & Rodr'iguez-Mart'inez, F. (2004). Developing a multi-agent system using MaSE and JADE. *Upgrade, 5*, 27–31.

Elghibari, F., Elouahbi, R., & el Khoukhi, F. (2019). Dynamic multi agent system for revising E-learning content materia. *The Turkish Online Journal of Distance Education, 20*(1), 131–144. https://doi.org/10.17718/TOJDE.522434

Giuffra, C., & Silveria, R. (2013). A multi-agent system model to integrate virtual learning environments and intelligent tutoring systems. *International Journal of Interactive Multimedia and Artificial Intelligence, 2*(1). https://doi.org/10.9781/ijimai.2013.217

Hamal, O., Faddouli, N.-E. el, & Harouni, M. H. A. (2021). Design and implementation of the multi-agent system in education. *World Journal on Educational Technology: Current Issues, 13*(4), 775–793. https://doi.org/10.18844/WJET.V13I4.6264

Holgado-Terriza, J., Pico-Valencia, P., & Garach-Hinojosa, A. (2020). *A Gateway for Enabling Uniform Communication among Inter-Platform JADE Agents*. https://doi.org/10.3233/AISE200027

Iglesias, C., Moreno Novella, J. I., Ricci, A., Diego, R. P., & Roman, D. (2020). A Gateway for enabling uniform communication among inter-platform JADE agents. ISBN 978-1-64368-090-3. In *Intelligent environments 2020 workshop proceedings of the 16th International conference on intelligent environments, 2020* (pp. 82–91), 82–91. Retrieved from https://dialnet.unirioja.es/servlet/articulo?codigo=7781879.

Jamili oskouei, D. R. (2014). Enhancing performance of learning management systems (LMSs) through intelligent agents. *MAGNT Research Report, 2*, 192–198.

Kelemen, V. (2006). *JADE TUTORIAL Simple Example for Using the JadeGateway Class*. Retrieved from https://jade.tilab.com/doc/tutorials/JadeGateway.pdf.

Kristensen, T., Bech, Ø., & Dyngeland, M. (2013). Towards a multi-agent E-learning platform. *Journal of Computer Engineering and Informatics, 1*, 64–81. https://doi.org/10.5963/JCEI0102004

Li, J.-W., Chang, Y.-C., Chu, C.-P., & Tsai, C.-C. (2012). A self-adjusting e-course generation process for personalized learning. *Expert Systems with Applications, 39*(3), 3223–3232. https://doi.org/10.1016/j.eswa.2011.09.009

Lopes, Y., Cortes, M., Gonçalves, E., & Oliveira, R. (2018). JAMDER: JADE to MULTI-agent systems development resource. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal, 7*, 63. https://doi.org/10.14201/ADCAIJ2018736398

Melesko, J., & Kurilovas, E. (2017). Personalised intelligent multi-agent learning system for engineering courses. In *2016 IEEE 4th workshop on advances in information, electronic and electrical engineering, AIEEE 2016 - proceedings* (pp. 1–6). https://doi.org/10.1109/AIEEE.2016.7821821

Morais, A. J., Oliveira, E., & Jorge, A. M. (2012). A multi-agent recommender system. *Advances in Intelligent and Soft Computing, 151*, 281–288. https://doi.org/10.1007/978-3-642-28765-7_33. AISC.

Morales-Rodríguez, M., Ramírez-Saldivar, J., Sánchez Solís, J., & Ramírez, A. (2012). Design of an intelligent agent for personalization of Moodle contents. *Research in Computing Science, 56*, 11–17. https://doi.org/10.13053/rcs-56-1-1

Nadrljanski, M., Vukic, D., & Nadrljanski, D. (2018). Multi-agent systems in e-learning. In *2018 41st International Convention on Information and communication technology, Electronics and microelectronics, MIPRO 2018 - proceedings* (pp. 990–995). https://doi.org/10.23919/MIPRO.2018.8400181

Neto, J. (2017). Multi-agent web recommender system for online educational environments. *Advances in Intelligent Systems and Computing, 619*, 309–312. https://doi.org/10.1007/978-3-319-61578-3_46

Rodríguez Marín, P. A., Duque, N., & Ovalle, D. (2015). Multi-agent system for knowledge-based recommendation of learning objects. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal, 4*(1). https://doi.org/10.14201/adcaij2015418089

Scutelnicu, A., Lin, F., Liu, T.-C., Graf, S., & McGreal, R. (2007). *Integrating JADE agents into Moodle*.

Selmi, A., Brahmi, Z., & Gammoudi, M. M. (2017). PACT: A new trust prediction method for multi-agents recommender systems. In *Proceedings - 2017 IEEE 26th International conference on enabling technologies: Infrastructure for collaborative enterprises* (pp. 9–14). WETICE 2017. https://doi.org/10.1109/WETICE.2017.29.

Trojahn, C., & Osorio, F. (2004, January). *Integrating Intelligent Agents, User Models, and Automatic Content Categorization in a Virtual Environment, 3220* pp. 128–139). https://doi.org/10.1007/978-3-540-30139-4_13

Vuković, I., Kuk, K., Čisar, P., Banđur, M., Banđur, Đ., Milić, N., et al. (2021). Multi-agent system observer: Intelligent support for engaged E-learning. *Electronics, 10*(12), 1370. https://doi.org/10.3390/ELECTRONICS10121370, 2021, Vol. 10, *Page 1370*.

Wooldridge, M. (2009). *An introduction to MultiAgent systems* (2nd ed.). John Wiley & Sons. Retrieved from https://www.wiley.com/en-gb/An+Introduction+to+Multi Agent+Systems%2C+2nd+Edition-p-9780470519462.