

# Intelligent Educational Assistant based on Multiagent System and Context-Aware Computing

## EducActiveCore: Experiment with Context-Aware and Multiagent Interaction

Fernão Reges dos Santos<sup>1</sup>, Pollyana Notargiacomo<sup>2</sup>  
Department of Post-Graduation in Electrical Engineering  
Mackenzie Presbyterian University  
São Paulo, Brazil

**Abstract**—This paper provides an overview of the current stage of EducActiveCore research, an orchestrated computational model, formed by different areas of artificial intelligent, combined to support personalized assistance to students in distance education process, mainly in interaction with Context-Aware environments. The Context-Aware environment applied in this research is observed in conjunction with IoT technologies. IoT is enabled by the latest developments in smart sensors, RFID, communication technologies, and Internet protocols. The basic premise is to have the resources availability and use arrangements, managed directly without human involvement to deliver a new class of smart environments for students. To support this central idea, a Multiagent model is proposed to assist students in interaction with context, determining autonomously the access to useful resources to students. This article introduces the overall research in progress and, the methods of an experiment tested with basic concepts of this scenario, implemented and used by a group of students in real locations. Results obtained during tests, indicates 93% of successful operation performed by this intelligent model on use prediction of resources and scheduling reservation.

**Keywords**—Context-aware computing; multiagent systems; artificial intelligence; internet of things

### I. INTRODUCTION

This article resumes the current state of one domains covered within the research in progress to apply intelligent computational models to support personalization of student's educational activities when interacting with context-aware environments. Added to an exponential growth of different data and media content, new scenarios are emerging on the horizon in analysis of behavior to student's computing interaction. Some of most recent evolving related areas stand out, like Internet of Things, Artificial Intelligence, Mobile Computing and Context-Aware Computing. Understanding the combination of these computing areas as complementary researches, this work investigates the applicability of these domains in a combined intelligent computational model named EducActiveCore to support online and remote assistance on personalization within educational process.

The content of this paper is organized in the following structure: The overall resume of EducActiveCore research and the background related is described in Section II. An introduction to Context-Aware Computing and Internet of Things are presented in Section III. Multiagent Systems and

Context-Aware Computing are introduced in a correlated domain in Section IV. The resume of computational models implemented to this experiment as its organization is presented in Section V. Method of experiment and preliminary results are presented in Section VI. Conclusions, issues of current stage and next phases of this research are introduced in Section VII.

### II. BACKGROUND AND RELATED WORK: INTELLIGENT COMPUTATIONAL MODEL EDUCACTIVECORE

In this section, we introduce the overall scope of the main research in progress and the focus of current work as an important implementation part within the proof of concept to applicability of proposed model.

#### A. Research Proposal and Objectives

The main objective in this research, considering its comprehensive scope, is established as a propose of an intelligent and adaptive computational model to support both educators and students through computer interaction within, initially, distance education process. Observing domains related to provide computational support in educational process, several challenges emerge regarding increase information available to educators during activities of task elaboration, collaborative process to collect and share information, interactivity between students and, hence, student's interaction with physical environments. The view of our main research proposing an intelligent computational model, named EducActiveCore, to cover these domains in a centric and self-orchestrated process, aims to act as intelligent assistant to help increase student's educational results since task elaboration (from educator's perspective) to activity execution by students.

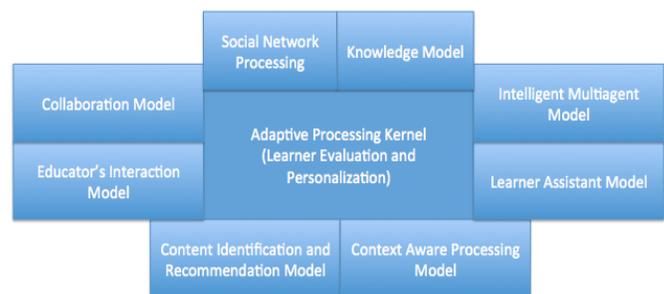


Fig. 1. Overview of model composition.

### B. Adaptive Processing Kernel

The orchestration proposed by EducActiveCore, contains a centric computational set of algorithms, structured as Adaptive Processing Kernel, which commands and interprets the complementary engines of processing.

The organization diagram contained in Fig. 1, illustrates a macro view of entire scope from research, organized in different computational engines. Computational engines identified by Processing in diagram (except for Kernel case) indicates a component completely accessory to this research vision, commonly, with its services accessed by EducActiveCore without its internal structure changed by this research. The engines identified by Processing Model, contains changes of behavior or complementary elements proposed in research scope. Finally, the engines with Model identification, contains its central behavior defined by research.

The scope of domains was structured in conceptual modules, organized according to its processing objectives with aim to reduce the complexity of Adaptive Processing Kernel that is the central engine proposed on this research. One of the main operations expected to this kernel is to identify the proper combination of student's mobility behavior with computing interaction and, execute the orchestration of the coupled complementary models to increase student's results on educational tasks.

The set of algorithms in this Kernel is organized to perform precisely mining and tracking on data collected from the complementary models. This set of algorithms is divided into two groups of distinct operation; Student's Evaluation and Student's Personalization. Basically, within Student's Evaluation operation, the algorithms collects student's data results related to educational tasks and determine their acceptance according to specific parameters. The group of algorithms responsible for Student's Personalization performs interactivity identification and student's assistance. Students can perform educational tasks in different physical environments and, hence, in simultaneous contexts. The Personalization processing identify the most recurrent class of data during interaction and determine if this interaction occurs in social network or a context-aware environment, for example.

As indicated in previous works [1], beyond the actual digital consuming profile of students with multi-connected and decentralized computer interaction behavior, moving student's attention from screens like Smart TVs, console games, tablets, smartphones, wearable and mobile devices, all those devices interconnected with internet services providing convergent media content, researches related to Internet of Things and Context-Aware computing are evolving rapidly.

Observing student's mobility and multiconnection possibilities from the perspective of environment navigation flow and, considering this flow as a learning object to be collected and processed, reinforces the need of fine personalization and characterization like previous researches covered [2]. Hence, this article presents the continuity of previous work [3] to elaborate the purpose of Adaptive Processing Kernel. In this precedent work, initial results were obtained from experiments using machine learning technics

(MLP Neural Networks) applied to classification tasks of environments and, an implementation of Multiagent protocol acting as assistant to route mapping and recommendation to physical locations. These results are data collected during the use of application implemented as proof of concept and feedback sent by students after experiment participation. Considering previous experiments and results obtained on research, this work evolves the proposed investigation of Adaptive Processing Kernel focusing on interoperability perspective correlating Context-Aware, Internet of Things and Multiagent system.

### III. CONTEXT-AWARE COMPUTING AND INTERNET OF THINGS: DEFINITION, MODELS AND LEARNING APPLICATIONS

#### A. Definition and Computing Models

Context-Aware computing can be described essentially by computational environments able to adapt itself and current context based on user presence and interactions, without explicit user supervision or intervention on this adaptive process [4]. There is no final and rigid boundary to specify what compose a context. Some authors refer to architecture key characteristics most appropriate to their research and field application to define resources and services of a context. Shared and common descriptions of these resources are presented as information describing and characterizing a person or object in interaction with computer application [5]. Beyond the informational aspects, physical elements and temporality of available resources [6] were published as features that are central to define a context.

Previous research (beginning of mobile research and development) resumed the context computing as a result on combination of three information domain regarding on where computational resources is, what kind of resources is provided and who is using it [7]. Modeling these characteristics in attributes, import and exchange it in form of relevant information to identify a context is a key process to support the context computing adaptive process [8]. The changing process of a context started by sensors, network events, mobile device location detection [9] or the combination of all these examples, provides several new different set of attributes to describe the context and related resources.

The Internet of Things (IoT) becomes an important study area since it denotes several impacts on local and global internet infrastructure and the services provided by computational environments [10]. The IoT bundles a variety of technologies (e.g. sensor hardware and firmware, communication technologies, semantic, cloud services, data modelling, storing, computational reasoning, high processing) and, eventually, combined together to establish its vision. The concept of IoT does not revolutionize or the existent fields of computing. It is evolving and complementing the Internet we already use.

The IoT holds the promise of establish a global network of useful and interoperable devices/resources supporting ubiquitous computing [11] and context-awareness among devices. Ubiquitous Computing and Context-awareness are key requirements of ambient intelligence, one of the main benefits considering the promises of the IoT [12]. IoT applied to

ambient intelligence would allow objects of common use to understand their environments, interact with users and make decisions [13]. A world full of smart objects holds several possibilities to improve business processes and user's lives and, hence, it also comes with threats and technical challenges that must be overcome. From perspective of users, deal with a huge variety and quantity of smart objects like sensors/devices, with its raw data generated, presents a challenge to project useful application and its viability [14]. Considering this perspective and the current research focus, we observe the educational application of IoT as a resource within context-aware environments operation.

### B. Context-Aware Learning Application

The variant set of IoT resources and its specific attributes within a context and, consequent different models of categorization and identification of actual resources and services content, requires specific observation and interaction regarding context computing and their applications. Previous works on context-aware computing applications within educational and mobile learning domains [15] helps to define the objective and scope of this computational model on this research. Investigations and development on context-aware applications to support language learning process can also be found on [16], using environment recognition to apply a compatible vocabulary according to user's location, supported by mobile devices and highlighting the importance of mobile devices to interact with context and evolving the application to detect the collaborative situational needs.

Another example based on context-aware computing implementing an instructional application [17] is built to identify tourist locations, relevant content and help tourist guides to take decisions considering the best matches for the group of visitors. Considerations of mobile device relevance within these tourist guide assistant and how embedding the solution on mobile devices to turn invisible and accessible to users were introduced.

On collaborative learning, previous research investigates learning environment built on a peer-to-peer architecture [18] and demonstrate the applicability of a common protocol (using meta-data structure) to determine context resources identification, information extraction and interoperability. The structure of meta-data, proposed in form of context-aware learning environment ontology, is processed with an adaptive model to evolve the ontology with aim to expand the interoperability. A range of supporting context-aware and IoT technologies to learning applications, including peripheral devices and sensors, image processing (recognition and augmented reality) can be found on [19].

### C. MultiAgent Systems and Context-Aware Computing

Commonly, Context-Aware Computing considers the self-managed configuration. Previous investigations with Agent systems applied to Context-Aware Computing, covered e-commerce business domains applying intelligent negotiation through case base reasoning and context history analysis [20]. Related to user context personalization, frameworks based on Internet webservices, Internet address identification and segmentation proposes frameworks to act on behalf of users on pre-selection and definition of context content and services

before turn it available to users [21]. Beyond the references related to relevant technical aspects to support multiconnected environments using agent-based model [22]. Previous researches covered Context-Aware Recommender Systems [23], investigating relevance of contextual information to recommend relevant content to users. The complementary research domain relevant to use on this current investigation is the Multiagent models applied to education. As precedent work, one specific approach proposed to Digital TV environment with adaptive content applied the combination of Multiagent model to educational content personalization [24]. In this work, as continuity of Adaptive Processing Kernel elaboration, we cover current step of research focusing the necessary behavior regarding to context-aware and Multiagent interaction aspects.

## IV. INTERACTION WITH CONTEXT-AWARE ENVIRONMENT ENABLED BY MULTIAGENT SYSTEM

### A. Multiagent Model as Interaction Protocol

For the purpose of this research, as a solution using context aware computational model, we identify as strongly necessary a local processing with representation of last set of actions taken and with minimal data used to compute the actual action and recommendation reflected on student's perspective.

The concept *Local* in this scenario means independent processing algorithms (not connected with Kernel) able to execute basic tasks based on last instructions dispatched by Kernel. This *Local* processing is foreseen to synchronize frequently on line with Kernel but ability to process disconnected emerges as necessary due to common issues related to complementary resources like Data Repository based on cloud platforms and its availability [25] or while a network connection is not available, for example. The aim of applying Multiagent Model in this research is to simplify the different local computation models. From student's utility view, the Agent Model can be available as embedded application on personal devices searching for available connections and, service discovering with context-aware environment according to student's profile and needs.

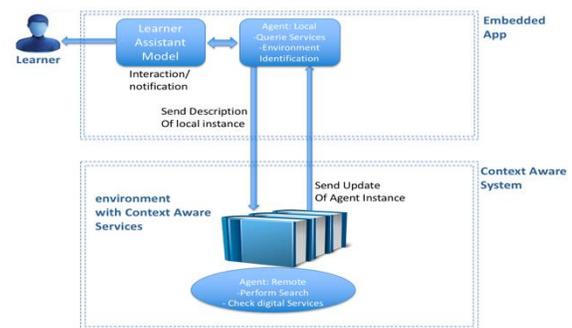


Fig. 2. Multiagent system as protocol interaction.

Fig. 2 illustrates the macro view of interaction through a Local instance on embedded application and its replica. The same Agent representation can be transported to an identified context-aware environment (in this case, acting as a protocol) and perform different computation using resources from environment instead of consuming local processing from

student's device. This strategy reduces the problems related to device's energy efficiency and other questions that could affect devices with embedded application containing Agent Model.

Considering this approach to use the student's local Agent as a protocol to be transported to context-aware environments, allows device's embedded application to send data related to student's profile and educational tasks where this student is involved, previous activities performed and its results, next activities and all other necessary data used by local Agent to execute its processing.

This set of data that represent the student's local Agent, creates another instance of Agent on context-aware environment. This new instance is described in this research as remote Agent, due its additional behavior to process Agent according to environment model using initial state received as input parameters. Specialization applied to each Agent model attends the variability aspects of context-aware environments where some attributes and behavior related to student's profile are shared to perform remote Agent tasks but considering environment characteristics, specific behavior and attributes are inherent of processing. Remote Agent, beyond the data provided by students and environment, can receive data from other Agent instances running in same context platform or allowing Learning Multiagent System [26]. It is useful to permit a structure of data communication to help agent identify unknown information, like other similar environments or the ranking level of environment services, indicated by other students. Tasks to be performed by remote Agent on context-aware environment can return to student's device as a result to update and merge behavior with device's agent instance and reflect its results on Kernel synchronization and Student Assistant Model.

In this experiment, we extended the mobile application implemented on prior experiments presented in Santos et al. [3] to add a reduced set of Multiagent operations regarding to synchronize with Kernel, simulate the content search on context-aware environment, collect additional information about context and performs scheduling reservation on selected resources.

### B. Context Aware Model - Service Discovery and Resource Identification

Considering the objectives of this research, the basic concept of context aware model must contain resources provided by a combination of physical location identification, descriptors and its digital services, observing the student's profile and needs. The model composition of services is variable and resources are intrinsically characteristic to environment accessed by student. A university laboratory can provide a 3D printer. A bookstore allows access on its book catalog. A commercial center indicates the store with specific scholar supplies. These distinct examples of environment differ on purpose and may vary on its characterization but they should present similarity with other environments under same objective classes.

Conceptually observing the local Agent behavior on Service Discovery and Resource Identification interaction, some common elements from context-aware platforms

definition emerges as necessary to insure the abstraction level and reduced complexity model applied to local Agent.

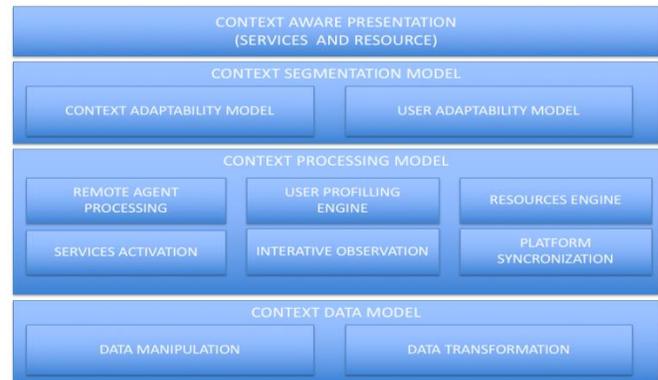


Fig. 3. Context aware platform organization.

Fig. 3 illustrates a macro model with minimal set of components and its layer organization proposed to establish a context-aware platform with based to this research vision.

The upper layer context aware presentation communicates with local Agent to present the Service Discovery and Resource Identification. The component Service Discovery provides the catalog of available services in environment. Resource Identification component permits the compatibility verification with a local Agent action intent related to a determined service previously discovered. Segmentation Model layer contains the management of user adaptability that control the level of interaction with student. The context adaptability uses this level of interaction as parameter to calibration and setup with student without interaction history. In example, if a student does not intend to share personal data with context, this combined engines process according to this definition. The layer Context Processing Model contains the engines responsible to process de dynamic of environment operation. In this layer we define the services orchestration and provisioning. Context Data Model layer contains the operations to access (query, transform and organize) data related to context and its resources.

In an overall and integrated example, if a local Agent queries and discovers an available 3D print service, resources engine verifies if the necessary drivers and supplies are compatible to attend the print request for student's project. If an incompatibility between service and requisition is detected, the resources engine could be update the available drivers to attend the request, if applicable. Eventually, a request for supplies reposition can be started if it turns insufficient. In case of impossibility with supplies reposition or driver's updates, the context segmentation configures the unavailability of this service specifically to user or group of users, and reflects it on presentation model to avoid new requests to same service.

## V. ENABLING COMPUTATIONAL RESOURCES, MIDDLEWARE IMPLEMENTATION AND CONTEXT-AWARE ENVIRONMENT

In this section, first we present the computational engines and its implemented operations involved in the scope of proof of concept implemented. To local Agent operation, the

following sequence of processing flow was implemented as initial behavior with the mobile application: 1) local Agent instance identify the context-environment through provided network WiFi name containing the defined pattern “CTX-ENV-NAME|ID”, where 'NAME|ID' pattern represents the the combination of environment and its unique identifier. 2) With local Agent connected to environment; firstly, application provides a copy of local Agent containing the current instance and all related attributes to desired services, and initiate the service request processing with context.

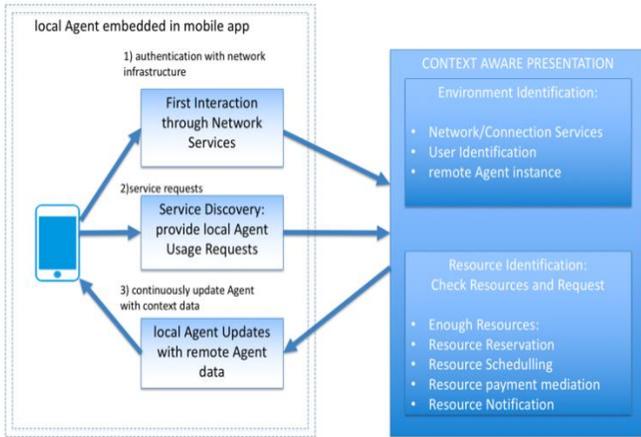


Fig. 4. Agent interactivity processing.

Fig. 4 contains the illustration of mobile application processing, embedding the Agent operations. Service Discovery and Requests are the main processing acting in environment resources. This block of operations executes the navigation on services catalog provided by environment and performs requests of resources reservation and content search according to desired needs. In a continuously flow, the mobile application receives an updated version of remote Agent and merges the local instance with new attributes and data received.

Fig. 5 contains an example of data received from context, representing the message describing the protocol of one remote Agent instance. This message contains group of data related to user’s profile, resources available on environment, resources reserved, beyond environment information and the attributes collection representing data shared by remote Agent instances of other students (agentShared indicative). It helps Multiagent learning process and increase the data information about environment during synchronization with Kernel.

With experimental objectives in simulate the basic operations of our proposed Context-Aware architecture, the middleware was implemented using base components from platform AWARE (<http://www.awareframework.com>). This open-source framework permits plugin development, extensions and complements, attending objectives of this initial experiment to provide an engine readily modifiable while investigation evolves and, consequently, new refinements and reviews on proposed architecture turns necessary adjust the model and the implemented platform.

```

{
  "environment": {
    "environmentID": "CTX-ENV-EletrLAB|Duz+3q6u/t6tyr7",
    "contextURI": "https://10.25.120.6/Duz+3q6u/t6tyr7",
    "contextClass": "GU"
  },
  "profile": {
    "agentProfileID": "3541601806C89387490B2CC08C528478",
    "studentName": "student 01",
    "studentEmail": "usr001@gmail.com",
    "lastKernelSync": "20171203134635"
  },
  "resourcesAvailable": {
    "printer3D": "Impressora 3d Creality Cr-10s 300x300x400",
    "contentCatalogCategories": [
      "eletronica profissional",
      "eletrica",
      "eletronica hobista"
    ]
  },
  "resourcesBooked": {
    "printer3D": true,
    "reservation": true,
    "availableSupplies": true,
    "scheduledTime": "17:00"
  },
  "agentShared": [
    {
      "agentSharedID": "C73BE44A6CF9C3AD21F48BE074E6762",
      "agentSharedUserTags": [
        "Facildadedeacesso",
        "perdometro",
        "materialparavenda",
        "protoboardmaissimples",
        "amploespacodetrabalho"
      ]
    }
  ],
  "contentCatalog": [
    "Arduino For Dummies",
    "Arduino: 101 Beginners Guide",
    "Arduino: 101 Beginners in 24 Hours",
    "Arduino Programming in 24 Hours",
    "Arduino Workshop: A Hands-On Introduction with 65 Projects",
    "Getting Started with Arduino: The Open Source Electronics Prototyping Platform",
    "Exploring Arduino: Tools and Techniques For Engineering Wizardry",
    "Arduino Robotics",
    "30 Arduino Projects for the Evil Genius",
    "Arduino: A Technical Reference"
  ]
}

```

Fig. 5. Segment of remote Agent data.

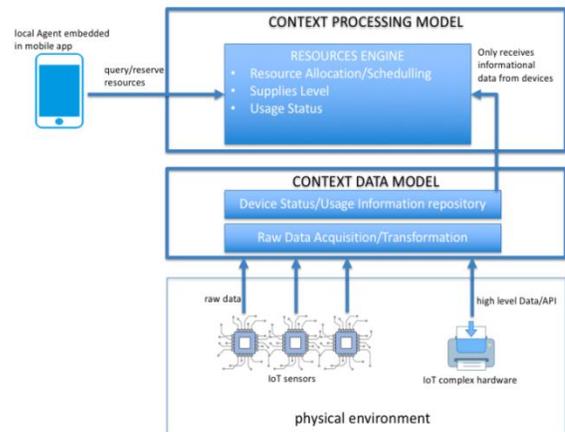


Fig. 6. Interaction Flow of Context and IoT.

Fig. 6 contains the illustration of layers and components organized to respond Agent’s requests. The lowest layer is responsible to integrate physical devices to platform. In case of IoT hardware sending only binary/raw data, without embedded application or driver/software providing interfaces to translate its state to consumer application, in this experiment we call it as IoT Sensor. This group of devices can contain temperature, presence or light sensors, providing their signals to component that performs data acquisition/transformation and turns this

data available to component responsible of data managing, both components in Context Data Model layer. To categories of hardware which embed any level of interfaces, drivers or operational software, we call it as IoT Complex Hardware. Within layer Context Processing Model, the implemented component Resources Engine contains the logic to handle local Agent requests to query services, status, supplies level and performs scheduling of usage reservations. This engine also manages remote Agent instance and performs the synchronization between agent instances.

## VI. EXPERIMENT WITH ACTIVITIES SUPPORTED BY MULTIAGENT INTERACTION AND CONTEXT-AWARE ENVIRONMENT

Since the intent of current stage of this research, requires identify the useful behavior of this computational engine proposed, a field investigation and simulated tests in real environments become important to map difficulties of real world application. This experiment was organized in the following steps:

- Find and select pre-defined physical locations with available devices prepared to act like IoT/environment resources as its services.
- To each physical location, the same base copy of middleware was installed, adjusted and parameterized according to its environment restrictions, number of devices and type of resources.
- A mobile application from a previous work was complemented to simulate the proposed Multiagent operations to perform interactions with Context-Aware environment, executing resources reservation and content search. This application also contains the operations of Multiagent synchronization with Adaptive Processing Kernel.
- A group of volunteer students tested the mobile application during real use of physical location services on educational tasks.
- Results and feedback collected to analysis.

### A. Experiment Method and Test Flow

The first list of physical locations invited to participate on this experiment, was prepared with co-work offices (<https://coworkingbrasil.org>), public libraries (<http://www.prefeitura.sp.gov.br/cidade/secretarias/cultura/bibliotecas/>) and private learning entities.

TABLE I. DISTRIBUTION OF PHYSICAL LOCATIONS

Locations	Invited	Interested	Participating	Volunteers
Co-Work Offices	60	3	0	0
Libraries	20	18	0	0
Learning Entities	20	17	6	21

Table I contains the categories distribution of location's list prepared. From a total of 100 locations contacted and invited to participate, 38 demonstrated interest on experiment. From this

group of 38 interested, 6 locations attended the criteria of this experiment on sharing data containing history of resources usage and having technological infrastructure/support to participate. Each physical location involved in this experiment, invited some users to test the mobile application during their educational activities, and shared a dataset with history of resources used by invited volunteer users. In this phase of research and, in this proof of concept, data regarding volunteer users are not profile identifiable to preserve participant's privacy.

The history of user's access on resources, provides data that was prepared (pre-processed) to train and test a dedicated layer of Multilayer Perceptron Neural Network within Adaptive Processing Kernel, to predict the next resource to be scheduled/provided to user. This layer is dedicated to this processing since for each user's operation of accept/reject a scheduling and content recommended by Kernel, a new cycle of this MLP Neural Network training is performed.

For each access on mobile application, hence, identifying an activity start by user, the local Agent starts to find to identify a new available context or connect to a new context. Once integrated with context, the local Agent performs its action function receiving from Kernel the resources reservation recommendation. Initially, an indicative with resource and agenda to reserve is received from Kernel.

A rule based on proximity of next available agenda is applied on local Agent as conflict resolution strategy [27], without Kernel synchronization to this task needed. In this operation, adapted to this experiment, a pre-reservation of resource occurs after Kernel recommendation and remains valid waiting to user's confirmation on the proposed agenda during a defined time by context.

Within Agent operation, a level of data sharing between Agent instances running on same environment occurs through remote Agent protocol, with a group of data indicating a collection of tags provided by users during the use of application. These tags are collected in context by local Agent and synchronized with Kernel to indicate attributes/properties related to environment that should be seen by every user in same environment. This strategy helps simulate a communication level with multiple instances of Agents that could take local action based on shared specific information. The last received data from Kernel, data collected from environment (device status for example) remains on local Agent and are synchronized with remote Agent on context. In parallel with remote Agent updates, the context manages devices status information (supplies availability and activity status) and performs scheduling operation.

Every operation executed by users on application are tracked to understand the navigation flow, including other previously implemented activities performed by application like routing recommendation that remains in this extended version.

### B. Preliminary Results

A total of 21 volunteers participated, using the mobile application during four weeks in support of their educational tasks interacting with the selected locations. Weekly, feedback

of each user and the data resulted from tests were collected, keeping same implementation of Agent model and middleware of context provided.

TABLE II. DATA FROM LOCATION WITH HIGHEST ACCES

Location A	Week 1	Week 2	Week 3	Week 4
Scheduling Recommended	34	46	71	74
Scheduling Accepted	12	37	62	69
Content Validated	3	21	26	43
Conflicts Solved by Agent	29	13	9	7

Table II contains the data collected from Location A, with highest number of application access by its users selected to this experiment. The evolution of model operating with this location can be observed with growing of recommendation acceptance.

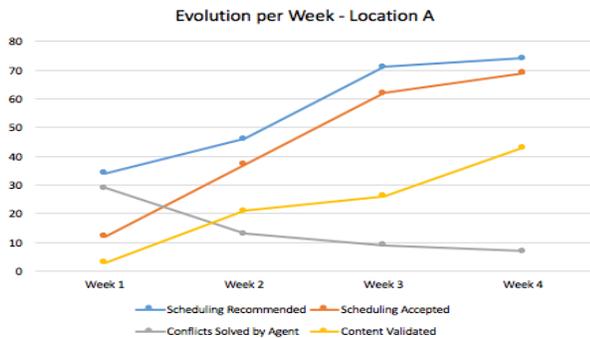


Fig. 7. Graphic representation of growing recommendation acceptance on Location A.

With this Location A can be observed an evolution from 35% of acceptance of Agent’s recommendation by users during activities on first week, to around 93% on fourth week.

TABLE III. DATA FROM LOCATION WITH LOWEST ACCESS

Location B	Week 1	Week 2	Week 3	Week 4
Scheduling Recommended	12	19	41	59
Scheduling Accepted	5	9	34	52
Content Validated	7	11	17	31
Conflicts Solved by Agent	4	7	5	4

Table III contains the data from Location B, with lowest access observed. With this Location B, at first week, evolution observed started from 41% of acceptance of Agent’s recommendation on first week, to around 88% on fourth week.

The graphic illustration indicating recommendation and acceptance in convergent tendency to Location A and Location B are contained in Fig. 7 and 8, respectively. Conflicts resolution can be observed as a decreasing event for both Locations presented.

After finished the period of four weeks of tests, the locations and the group of volunteers sent final feedbacks about their participation on this experiment.

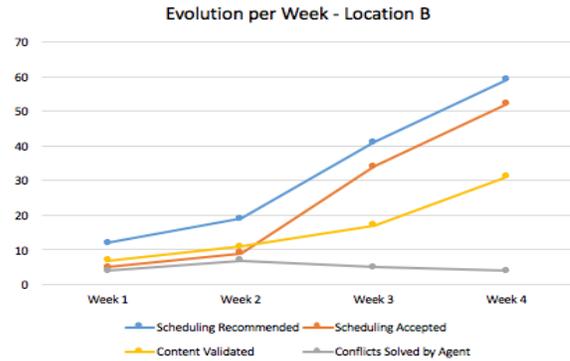


Fig. 8. Graphic representation of growing recommendation acceptance on Location B.

### VII. CONCLUSION AND FURTHER WORK

Lessons learned selecting locations, indicates the need of a study to investigate privacy and technology aspects to each category of location. During investigation on details to map the reasons of lower interest from Co-Work Offices on experiment, a questionnaire answered reveals business concerns on avoid sharing data of services usage and, eventually, negative ratings by users.

To libraries category, the public institutions invited to participate presented a high number of interested locations, however, none of institutions contacted owns the necessary infrastructure to participate on tests. Some of these libraries, sharing a public platform to manage the book collection does not have an automated control over other resources provided like room reservation and digital/media resources. Given this scenario, an open space emerges to an intelligent model as proposed in this research to complement an automated platform applied to this environment category.

The group of Learning Institutions participating in this experiment, essentially centers with Arduino/IoT specialization courses, presented the necessary requirements and improvements in its environment to participate, hence, totality of volunteers to test. These spaces with intrinsic technology profile can be useful to understand the necessary complement to scale the research model to other environment categories.

Considering the main premise of this research about a modular intelligent computational engine, this experiment preparation with setup and customizing a framework of context middleware, facilitates the process and indicates a fluid strategy to evolve the model proposed and future tests.

The modeling of the experiment indicates the context architecture proposed, acting as a computational tool to manage and mediate the access to the IoT devices of the environment, as favorable strategy. This architecture enables formation of data repositories that can be accessed when needed by Kernel and reduce the complexity of managing multiple devices.

Owing to the space limit and the focus of this work, many complementary aspects covered during research could not be discussed in details here. This includes Multiagent self-sustainability that may be applied to maintaining stability of the system (with respect to an appropriate size of the population of agents), the capability to respond and attend to the emergency situations occurring in the environment (breakdown hardware/software of infrastructure), as well as in the environment and user requirements (e.g. change of computation parameters or devices/resources coupled). Not detailed in this resume, the Neural Networks layer added to Kernel, its topology and data normalization method used on prediction process process of resources reservation, followed the basic method presented by Dos Santos et al. 2017 [3].

Feedback collected with volunteers during the tests, pointed desirable features on future version of model. About 80% of volunteers indicated automated activation of resources and communication of execution status as important improvement. Around 68% of volunteers indicated as necessary, the feature of previous software/drivers compatibility check to access electronic resources. As desirable resource, about 90% of volunteers indicated active recommendation of new location, added to resources scheduled reservation in case of unavailability/error on current location resource.

The next stage of research will cover the computational engine model responsible to process and predict the influence of resources on activities results.

Aspects of educator's perspective, regarding to support intelligent task composition is one of next research phase to cover.

Conclusions initially obtained of this current state of research provides basement to evolve the investigation and future experiments.

#### REFERENCES

- [1] Ally, Mohamed. Mobile learning: Transforming the delivery of education and training. Athabasca University Press,
- [2] Silva, Luciano et al. Using conceptual lattices to represent fine granular learning objects through scorm meta-objects. The Electronic Journal of e-Learning, v. 4, n. 2, p. 141-148, 2006.
- [3] Dos Santos, Fernao Reges; Notargiacomo, Pollyana. EducActiveCore: Computational Model to Educational Personalization Based on Multiagent and Context-Aware Computing. International Journal of Learning, Teaching and Educational Research, v. 16, n. 11, 2017.
- [4] Baldauf, Matthias; Dustdar, Schahram; Rosenberg, Florian. A survey on context-aware systems. International Journal of Ad Hoc and Ubiquitous Computing, v. 2, n. 4, p. 263-277, 2007.
- [5] Abowd, Gregory D. et al. Towards a better understanding of context and context-awareness. In: Handheld and ubiquitous computing. Springer Berlin Heidelberg, 1999. p. 304-307.
- [6] Bellavista, Paolo et al. A survey of context data distribution for mobile ubiquitous systems. ACM Computing Surveys (CSUR), v. 44, n. 4, p. 24, 2012.
- [7] Schilit, Bill; Adams, Norman; Want, Roy. Context-aware computing applications. In: Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on. IEEE, 1994. p. 85-90.
- [8] Makris, Prodromos; Skoutas, Dimitrios N.; Skianis, Charalabos. A survey on context-aware mobile and wireless networking: On networking and computing environments' integration. Communications Surveys & Tutorials, IEEE, v. 15, n. 1, p. 362-386, 2013.
- [9] Ogata, Hiroaki; Yano, Yoneo. Context-aware support for computer-supported ubiquitous learning. In: Wireless and Mobile Technologies in Education, 2004. Proceedings. the 2nd IEEE International Workshop on. IEEE, 2004. p. 27-34.
- [10] Al-Fuqaha, Ala et al. Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, v. 17, n. 4, p. 2347-2376, 2015.
- [11] Bandyopadhyay, Soma et al. Role of middleware for internet of things: A study. International Journal of Computer Science and Engineering Survey, v. 2, n. 3, p. 94-105, 2011.
- [12] Dohr, Angelika et al. The internet of things for ambient assisted living. In: Information Technology: New Generations (ITNG), 2010 Seventh International Conference on. Ieee, 2010. p. 804-809.
- [13] Braun, Andreas; Wichert, Reiner; Maña, Antonio. Ambient Intelligence. Springer International Publishing AG, 2017.
- [14] Lee, In; Lee, Kyoochun. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. Business Horizons, v. 58, n. 4, p. 431-440, 2015.
- [15] Huang, Yueh-Min; Chiu, Po-Sheng. The effectiveness of a meaningful learning-based evaluation model for context-aware mobile learning. British Journal of Educational Technology, v. 46, n. 2, p. 437-447, 2015.
- [16] Cheverst, Keith et al. Developing a context-aware electronic tourist guide: some issues and experiences. In: Proceedings of the SIGCHI conference on Human Factors in Computing Systems. ACM, 2000. p. 17-24.
- [17] Yang, Stephen JH. Context aware ubiquitous learning environments for peer-to-peer collaborative learning. Journal of Educational Technology & Society, v. 9, n. 1, p. 188-201, 2006.
- [18] Kwon, Oh Byung; Sadeh, Norman. Applying case-based reasoning and multiagent intelligent system to context-aware comparative shopping. Decision Support Systems, v. 37, n. 2, p. 199-213, 2004.
- [19] Traxler, John; Kukulska-Hulme, Agnes (Ed.). Mobile learning: The next generation. Routledge, 2015.
- [20] Kwon, Ohbyung; Choi, Sungchul; Park, Gyuro. Nama: A context-aware multiagent based web service approach to proactive need identification for personalized reminder systems. Expert Systems with Applications, v. 29, n. 1, p. 17-32, 2005.
- [21] Soldatos, John et al. Agent based middleware infrastructure for autonomous context-aware ubiquitous computing services. Computer Communications, v. 30, n. 3, p. 577-591, 2007.
- [22] Aouatef, Chaib; Iman, Boussebough; Allaoua, Chaoui. Adaptive composition of services in context-aware ambient intelligent systems. In: Proceedings of the 2014 International C\* Conference on Computer Science & Software Engineering. ACM, 2014. p. 26.
- [23] Adomavicius, Gediminas; Tuzhilin, Alexander. Context-aware recommender systems. In: Recommender systems handbook. Springer, Boston, MA, 2015. p. 191-226.
- [24] Santos, F. R. ; Mustaro, P. N. . Educator: A Tutoring System Based On Multiagent Approach And Social Network Analysis For Development A Digital Tv Environment with Adaptive And Collaborative Proposal. In: 23rd ICDE World Conference on Open Learning and Distance Education including the 2009 EADTU Annual Conference, 2009, Maastricht. Book of Abstracts, 2009.
- [25] Alshwaier, Abdullah, Ahmed Youssef, and Ahmed Emam. "A new Trend for E-learning in KSA using educational clouds." Advanced Computing 3.1 (2012): 81.
- [26] Weiss, Gerhard (Ed.). Multiagent systems: a modern approach to distributed artificial intelligence. MIT press, 1999 - p241-278.
- [27] Cicirelli, Franco; Giordano, Andrea; Nigro, Libero. Efficient environment management for distributed simulation of large-scale situated multi-agent systems. Concurrency and Computation: Practice and Experience, v. 27, n. 3, p. 610-632, 2015.