

# Ubibot: Agente Inteligente Consciente do Contexto de Aprendizagem do Usuário Integrado ao Ambiente Moodle

Leo Natan Paschoal

Universidade de Cruz Alta – Centro  
de Ciências Humanas e Sociais  
+555533211563  
leonatanpaschoal@gmail.com

Patricia Mariotto Mozzaquatro  
Chicon

Universidade de Cruz Alta – Centro  
de Ciências Humanas e Sociais  
+555533211563

pmozzaquatro@unicruz.edu.br

Aliane Loureiro Krassmann

Universidade Federal do Rio Grande  
do Sul – Centro de Novas  
Tecnologias na Educação  
+555133083986

alkrassmann@gmail.com

Manuel Osorio Binelo

Universidade Regional do Noroeste  
do Estado do Rio Grande do Sul  
+555533211563

manuel.binelo@gmail.com

## ABSTRACT

This article presents a proposal for the development of an educational tool that aims to help Software Engineering's discipline's students, considering the actual context in which learning takes place. The proposal follows the ideology of providing education from a level of ubiquity, where an intelligent agent is inserted into a virtual environment and its feedback is delivered in a manner oriented to student's level of knowledge and performance. It is considered in this research attributes of computational intelligence inserted in the agent from Q-learning algorithm, which aims to recognize states that represent the user's context, providing a learning based on reinforcement.

## RESUMO

O presente artigo apresenta uma proposta para o desenvolvimento de uma ferramenta educacional que tem como objetivo auxiliar o estudante na disciplina de Engenharia de *Software*, considerando o contexto real em que a aprendizagem ocorre. A proposta segue a ideologia de prover educação a partir de um nível de ubiquidade, em que um agente inteligente é inserido em um ambiente virtual e seu *feedback* é emitido de maneira orientada ao nível de conhecimento e desempenho do estudante. Considera-se nesta pesquisa atributos de inteligência computacional inseridos no agente a partir do algoritmo *Q-learning*, o qual visa reconhecer estados que representam o contexto do usuário, originando um aprendizado baseado em reforço.

## Categories and Subject Descriptors

K.3.2 [Computer and Information Science Education]:  
Information systems education – Computer Science Education  
and Information Systems Education

## General Terms

Experimentation.

## Keywords

Intelligent agents. Ubiquitous learning. Context-aware services.  
Electronic learning.

## 1. INTRODUÇÃO

O vislumbre de computadores onipresentes vem fazendo os sistemas de informação adotarem metodologias e técnicas computacionais para identificar características humanas e informações sobre os usuários. Observa-se um crescente número de pesquisas que possuem como objetivo a implementação de ambientes inteligentes que identificam o contexto do usuário e se adaptam em função da informação coletada. As interfaces gráficas para usuários em ambientes ubíquos são vantajosas em termos de interação. Neste sentido, surgem na literatura científica estudos sobre a adaptação da interface de ambientes virtuais de aprendizagem a partir de informações referentes ao usuário, tais como: estilo cognitivo [1, 2], aspectos culturais [3], nível de conhecimento [4], estilo navegacional [5], entre outras; e informações tecnológicas referentes ao dispositivo, tal como: a velocidade de conexão com a Internet do dispositivo [2, 6].

A experiência e o interesse do estudante são fatores importantes quando utilizados com propósito educacional, podendo garantir o sucesso acadêmico [7]. Sob esta perspectiva, utilizar as preferências dos estudantes para apresentar materiais didáticos apropriados às suas características e ocultar os materiais que não são direcionados ao seu perfil, são características de ambientes virtuais de aprendizagem ubíquos, uma vez que os objetos educacionais são disponibilizados ao estudante considerando o seu contexto [8].

A criação de sistemas educacionais dinâmicos, intuitivos e adaptativos é explorada em pesquisas atuais [9]. Visa-se construir ambientes em que a interação seja pouco intrusiva, para que os estudantes possam se concentrar em atividades consideradas importantes, como o contato direto com materiais didáticos. Neste sentido, ambientes educacionais ubíquos permitem que o usuário estude sempre, independente de lugar e dispositivo [8], resultando em uma aprendizagem mais autônoma.

Nesse sentido, enquanto há pesquisas que abordam a adaptação dinâmica de ambientes ao perfil do usuário [2,3,5], contribuindo com a *ubiquitous learning*, também existem estudos que relatam o uso de agentes inteligentes como recurso tecnológico para oferecer um processo de aprendizado mais atrativo ao estudante [10, 11] e conseqüentemente, fazendo com que o seu desempenho

possa progredir. No estudo de [10] foi evidenciado que um agente com atributos de conversação contribuiu com a redução da carga cognitiva do material didático. A pesquisa de [12], por sua vez, mostra que agentes inseridos em ambientes educacionais podem interagir de maneira antropomórfica com o estudante. Desse modo, observa-se que agentes podem tornar o ambiente virtual mais dinâmico [9].

Agentes com capacidade de conversação são conhecidos na literatura como *chatbots* ou robôs de conversação. Eles são desenvolvidos com técnicas de inteligência artificial, as quais permitem simular a conversa com seres humanos [13], através de mensagem de texto e voz [10]. Quando esses agentes são inseridos em ambientes educacionais, recebem a nomenclatura de agentes pedagógicos animados. Salienta-se que independente de nomenclatura, tanto agentes conversacionais quanto agentes pedagógicos podem ser considerados agentes inteligentes, pois utilizam sensores para identificar o seu ambiente e atuadores para agir sobre ele. No caso de um agente conversacional, ele percebe o seu ambiente através de uma sequência de teclas digitadas pelo usuário, e o meio como o agente atua sobre o ambiente é realizado por intermédio de linguagem natural.

Estudos realizados no contexto de *chatbots* abordam a construção da base de conhecimento desses agentes de maneira restrita a uma determinada área de conhecimento e a um ambiente virtual específico [13]. Por exemplo, o agente ELAI é capaz de auxiliar estudantes de Redes de Computadores que estejam utilizando o ambiente TCN<sup>5</sup> [11], já o agente AGEbot é capaz de responder aos questionamentos dos usuários de um Sistema Tutor Inteligente, sobre o assunto de Epilepsia [14].

Ao construir a base de conhecimento de um agente inteligente, o desenvolvedor deve elaborar uma base de dados considerando as possíveis interações que o agente será capaz de responder. Entretanto, não é possível prever todas as possíveis ações de um usuário. Por esse motivo, *chatbots* normalmente são limitados [13]. Para resolver essa limitação, surge o método de aprendizagem por reforço, em que o agente inteligente aprende por sucessivas interações com o ambiente onde está inserido. Um agente inteligente apoiado em aprendizagem por reforço aprenderá e evoluirá com o auxílio do ambiente [14].

O advento das tecnologias de informação e comunicação permitiu que os agentes inteligentes fossem executados via dispositivos móveis [15], assim como o acesso a ambientes virtuais de aprendizagem [2, 6]. Considerando o fato de que ambientes virtuais são capazes de identificar o contexto do usuário, é possível desenvolver um agente inteligente com a capacidade de coletar informação sobre o estudante.

Ao construir um agente que auxilie o usuário considerando o seu contexto e possua um grau de mobilidade, ou seja, se adapte a diferentes interfaces de dispositivos móveis, é possível desenvolver um agente inteligente com um nível de ubiquidade.

Diante do exposto, este artigo visa apresentar o desenvolvimento de um agente inteligente que suporta a comunicação com o usuário via mensagem textual, considerando o contexto do usuário. O agente chama-se Ubibot, uma aglutinação entre as palavras *chatbots* e *ubiquitous learning*. Ele é capaz de identificar o nível de conhecimento e desempenho dos estudantes matriculados na disciplina de Engenharia de *Software*, considerando as notas das atividades realizadas na disciplina através do ambiente virtual Moodle, adequando as suas

mensagens a este contexto e permitindo que o estudante realize suas atividades na disciplina de forma mais atenta e direcionada. Desse modo, o Ubibot observa o contexto do usuário e atua de maneira transparente ao estudante, simulando um tutor pedagógico de conteúdos.

Visando reconhecer as possibilidades dessa proposta, no decorrer deste artigo são descritos alguns experimentos realizados para a efetiva implementação do agente Ubibot.

## 2. AMBIENTES VIRTUAIS DE APRENDIZAGEM UBÍQUOS

Os ambientes virtuais de aprendizagem (AVA) são utilizados no ensino presencial, semipresencial e a distância, com função de prover recursos para o processo de ensino e aprendizagem. Os recursos são variados e dependem da arquitetura e finalidade do sistema, mas assemelhassem-se por possuírem ferramentas de comunicação (síncrona e assíncrona), informação e gestão. Possuem ferramentas que permitem a comunicação entre aluno, professor e tutor, e também oferecem diferentes tipos de instrumentos para o suporte a criação de textos colaborativos, *upload* de arquivos, enquetes e questionários, entre outros recursos, os quais permitem a troca de materiais entre aluno e professor [4]. Por fim, pode-se dizer que os AVA permitem a elaboração de cursos totalmente *online*.

Conforme dados obtidos por pesquisa realizada no ano de 2013, os AVA mais utilizados por instituições brasileiras públicas de ensino, são: Moodle, Atutor, Aulanet, Blackboard, TelEduc e WebCT [16]. Ressalta-se que algumas instituições utilizam mais de um ambiente de apoio a aprendizagem. Em 98% dos casos o Moodle é o mais utilizado. Em um mapeamento sistemático realizado em 2014, sobre teses de doutorado relacionadas a AVA, o Moodle foi identificado como objeto de pesquisa em 20% dos casos, destacando-se como o ambiente mais utilizado nestes tipos de estudos [17]. Ainda, o Moodle destaca-se por possuir interface personalizável a dispositivos móveis, garantindo que o acesso ocorra em qualquer dispositivo [9].

A aprendizagem independente de dispositivo originou-se da difusão das tecnologias de processamento e comunicação dados em sistemas de apoio a aprendizagem. Inicialmente, os AVA eram disponibilizados apenas para *desktops*, o que resultava em uma aprendizagem centrada e dependente, conhecida como *e-learning*, em que a aprendizagem ocorre por meio de computador e o usuário fica limitado a um espaço físico para aprender.

Em estudos recentes, discute-se a adaptação de ambientes virtuais de aprendizagem ao contexto do usuário [18]. Busca-se desenvolver ambientes ubíquos, idealizando a proposta da computação ubíqua vislumbrada por [19]. Entretanto, a tecnologia existente atualmente permite a implementação de ambientes com um nível de ubiquidade, mas não totalmente ubíquos [9]. Portanto, pesquisas referentes a sistemas educacionais, buscam utilizar sensores virtuais e/ou físicos para capturar informações sobre os utilizadores dos ambientes, e técnicas como hipermídia adaptativa para modificar as interfaces e conteúdos desses ambientes ao contexto identificado, originando ambientes virtuais de aprendizagem ubíquos.

De acordo com [20], as pessoas são diferentes e por isso aprendem em ritmo e maneira diferente. Com base nesta perspectiva, ambientes que utilizam a tecnologia da computação ubíqua permitem que a aprendizagem seja personalizável ao

contexto do estudante. Em aprendizagem ubíqua, o usuário aprende com o material didático apropriado, por meio de um ambiente adaptado a suas ações e necessidades. Nesse sentido, torna-se importante aproveitar características de aprendizagem inerentes ao estudante para apresentar um ambiente personalizado. Assim, surgem na literatura pesquisas que utilizam o nível de conhecimento do usuário para ofertar ambientes de *ubiquitous learning* [4].

Aspectos culturais podem influenciar no aprendizado de estudantes [3]. A partir desse pressuposto, [3] adicionou uma extensão ao ambiente hipermídia AdaptWeb, o qual apresenta o conteúdo direcionado ao usuário. O ambiente AdaptWeb é provido de diversos recursos para a aprendizagem com um nível de ubiquidade. Estudos foram realizados a fim de tornar o ambiente orientado a diversos contextos, como estilo de aprendizagem, habilidade do estudante e aspecto cultural.

Enquanto que o AdaptWeb é personalizável a diversas particularidades do aluno, o Moodle oferece suporte limitado em termos de adaptação de conteúdo [5]. Este AVA destaca-se em nível de adaptação de interface para o contexto dos dispositivos móveis, sendo o seu suporte à *mobile learning* destacado em pesquisas [1]. O sistema suporta a integração do *framework* Bootstrap, utilizando a tecnologia *web responsive* para adaptar-se aos diversos tamanhos de tela dos dispositivos.

No entanto, estudos apontam para a adaptação dos AVA aos diferentes contextos do usuário [2,9]. Desse modo, é desejável que o Moodle possa oferecer materiais direcionados ao usuário, incorporando, além de características de *mobile learning*, a sensibilidade ao contexto [6].

Ao considerar os diferentes recursos do ambiente Moodle, acesso ao código fonte e forte adaptabilidade a diferentes *ecrãs*, [2] utilizou módulos que realizam o diagnóstico do estilo cognitivo do estudante e identificam a velocidade de conexão à Internet do dispositivo para adaptar o ambiente Moodle à computação ubíqua, voltando-o à *ubiquitous learning*.

Essa seção objetivou apresentar uma contextualização sobre AVA com tecnologia de computação ubíqua. A *ubiquitous learning* permite que o estudante aprenda de forma personalizada, considerando características, necessidades e preferências do usuário ao apresentar os conteúdos. Observa-se que ambientes de *e-learning* como o AdaptWeb e o Moodle, possuem versões que se personalizam a diferentes contextos e permitem que o aprendizado do usuário seja móvel e ubíquo, impulsionando uma proposta de aprendizagem dinâmica. Nesse âmbito, surgem os agentes com atributos de inteligência artificial para tornar a aprendizagem mais atrativo aos estudantes.

### 3. AGENTES INTELIGENTES

Uma das vertentes da inteligência artificial está centrada nos agentes inteligentes [21]. Mas o que são agentes inteligentes? Existem diversas abordagens sobre a definição concreta do seu significado. A mais aceita é que são entidades que utilizam sensores para perceber e atuadores para agir sobre um determinado ambiente [21]. Agentes inteligentes realizam operações em favor do usuário com um grau de independência ou autonomia, sendo capazes de conhecer ou representar os objetivos do usuário [22].

Na literatura encontram-se vários conceitos sobre agentes. Alguns estudos consideram que estes possuem crenças e intenções [23].

Outros enfatizam que um programa deve possuir algumas propriedades para ser considerado um agente inteligente, tais como: autonomia, pró-atividade, reatividade, continuidade temporal, capacidade social/comunicabilidade, capacidade de adaptação, mobilidade e flexibilidade [22].

Porém, um agente inteligente não precisa possuir todas as propriedades, dependendo da sua finalidade [10]. Desse modo, uma ou mais propriedades são suficientes para determinar que o agente possua um grau de inteligência. Nessa pesquisa parte-se do pressuposto de que para um agente ser considerado inteligente ele deve utilizar métodos de inteligência artificial para atingir seus objetivos [24]. Ao atingir um objetivo, o agente pode enquadrar-se no conceito [21], em que um agente que age sobre um ambiente para atingir um melhor resultado é considerado um agente racional.

Os agentes recebem diversas nomenclaturas, de acordo com sua finalidade. Agentes com capacidade de conversar com seres humanos são conhecidos como agentes conversacionais ou *chatterbots*. Estes agentes possuem duas propriedades que podem considerá-los como agente inteligente. Uma delas é a capacidade social, em que o agente é capaz de comunicar-se com os seres humanos, outra é a pró-atividade, em que o agente toma iniciativa para satisfazer o seu objetivo, que é atender o usuário.

*Chatterbots* são desenvolvidos para propiciar um processo de ensino e aprendizagem mais eficiente e interativo, visando ser um instrumento tecnológico complementar para o aprendizado [24]. Estudos diversificados foram realizados em torno dos *chatterbots*, sendo que, a maioria dos pesquisadores utilizam os *chatterbots* no contexto educacional devido à sua função de oferecer instruções personalizadas ao estudante, aumentando a motivação ao estudar [11].

O desenvolvimento de um agente conversacional normalmente é realizado por intermédio de linguagem de marcação AIML (*Artificial Intelligence Markup Language*), um recurso que permite o armazenamento de pares conversacionais formulados em linguagem natural, os quais representam o domínio de conhecimento do *chatterbot* [24], elucidado por perguntas e respostas pré-definidas pelo desenvolvedor do agente. Neste sentido, o desenvolvedor em AIML deve programar as perguntas que podem ser solicitadas pelo usuário e as respostas utilizadas pelo agente.

Para um agente simular conversas com seres humanos, ele necessita de um motor de inferência [25]. Este é responsável por reconhecer os padrões informados pelo usuário e retornar a resposta adequada a este padrão. Basicamente, o motor interpreta a pergunta digitada pelo usuário e busca na base de conhecimento uma unidade que contenha a resposta adequada para a pergunta. Desse modo o agente, ao ser questionado, retorna uma resposta que se aproxima da expectativa do usuário [10].

A linguagem AIML, usualmente utilizada em *chatterbots*, foi criada especificamente para mecanismos de conversação [26]. Alguns estudos utilizam a plataforma online Pandorabots<sup>1</sup> [24], outros a Program O<sup>2</sup>, pois permitem a criação e armazenamento de base de conhecimento do agente, além de possuírem um motor de inferência capaz de interpretar essa base de conhecimento.

<sup>1</sup> Disponível em <http://www.pandorabots.com/>

<sup>2</sup> Disponível em <http://www.program-o.com/>

Ambos são ambientes que apresentam recursos para criar e publicar *chatterbots*.

Alguns estudos distinguem agente conversacional de agente inteligente. Pode ser visto a distinção clara no estudo de [12]. Entretanto, ao considerar outros estudos, um agente conversacional apresenta recursos de agente inteligente [11]. Conforme [21], um agente precisa ser inteligente para executar suas tarefas, pois com inteligência as realiza de maneira autônoma. Salienta-se que um agente conversacional não possui a propriedade de autonomia, porque depende de um usuário para interagir, além de não ser capaz de adaptar-se ao ambiente e não apresentar a propriedade de flexibilidade, ou seja, não considerar o estado em que o ambiente se encontra.

Busca-se neste estudo atribuir métodos de inteligência artificial em um agente conversacional a fim de dá-lo autonomia. Desse modo, alia-se a esta proposta os estudos de [21] e [22], assim como, a diversas pesquisas que utilizam *chatterbots* no contexto educacional [10,11,15,24], visando oferecer uma ferramenta tecnológica para apoiar o estudante.

Os autores [13] dissertam sobre a base de conhecimento limitada de *chatterbots*, visto que o criador do agente deve modelar todas as possíveis interações. Logo, a área de domínio de um *chatterbot* normalmente é restrita. Como proposta para solucionar esta limitação, [14] descrevem sobre o uso de aprendizagem por reforço, em que o *chatterbot* evolui com auxílio do ambiente. Desse modo, pode ser considerado um agente inteligente, visto que possui aspectos de autonomia.

Esta seção teve como objetivo contextualizar os agentes inteligentes, principalmente os que possuem a propriedade de comunicabilidade, denominados aqui como agentes conversacionais/*chatterbots*. Observa-se que existem pesquisas que consideram os *chatterbots* como agentes inteligentes e outras não consideram, pois os *chatterbots* não apresentam autonomia. Também foi possível constatar o uso dos *chatterbots* no contexto educacional, visto que estes agentes podem oferecer uma aprendizagem interativa ao estudante.

Considerando algumas limitações dos *chatterbots*, é possível identificar a demanda por pesquisas que consideram o uso de técnicas de inteligência artificial para dar autonomia ao agente, especificamente providas de aprendizagem de máquina, como a aprendizagem por reforço.

#### 4. APRENDIZAGEM POR REFORÇO

A aprendizagem de máquina parte do pressuposto de que não é possível prever todas as situações possíveis que um agente pode encontrar [21]. Existem basicamente três tipos de *feedback* para aprendizagem de máquina, sendo eles: aprendizagem não supervisionada, aprendizagem por reforço e aprendizagem supervisionada. Nesta seção é realizada uma discussão sobre a técnica de aprendizagem por reforço, em que um agente aprende por sucessivos reforços.

A técnica de aprendizagem por reforço permite que um agente identifique o estado em que se encontra em um ambiente. Nesse sentido, possui a capacidade de adquirir conhecimento sobre o ambiente onde está inserido [27], ao realizar ações e observar as mudanças causadas por essa ação. Ao observar essas transições de estados, adquire experiência.

Assim, aprendizagem por reforço consiste em uma estrutura composta de estados, ações e recompensa, sendo que a recompensa é o reforço adquirido [28]. Desse modo, é condizente utilizar um algoritmo que obtenha a melhor ação, visto que ela resulta em um melhor reforço. Observa-se a existência de diversos tipos de algoritmos de aprendizagem por reforço, como *Average Reward Reinforcement Learning*, *Actual Return Reinforcement Learning*, *Dyna-Q*, *Feudal Q-learning*, *Compositional Q-learning*, *Hierarchical Q-learning*, *Recurrent Q-learning*, *R-learning* e *SARSA*. Muitos desses mencionados são derivados do *Q-learning*, por ser um dos primeiros algoritmos para este tipo de aprendizagem e um dos mecanismos mais relevantes propostos para solução de problemas [29].

O *Q-learning*, também conhecido como algoritmo de aprendizagem Q, é utilizado para definir a melhor ação a ser tomada por agentes, entre algumas alternativas. Trata-se de um algoritmo de aprendizagem por reforço que aprende o valor  $Q(s,a)$  em cada estado ( $s$  – *state*) por intermédio de cada ação ( $a$  – *action*) [30]. Seu funcionamento consiste em basicamente em três passos: a) o agente realiza uma ação ( $a$ ) em um estado ( $s$ ); b) o agente verifica a consequência/recompensa ( $r$ ) da ação realizada ( $a$ ); e c) por fim, o agente estimula o valor do estado que a ação o levou [27].

O algoritmo permite identificar as melhores ações que devem ser tomadas em determinados estados de um ambiente por um agente. Assim, o *Q-learning* permite que o agente tenha autonomia para realizar determinadas tarefas, visto que irá saber o que ocorre em cada particularidade do ambiente.

Neste estudo, pretende-se desenvolver um *chatterbot* que possa experimentar autonomamente o ambiente onde está inserido, utilizando o algoritmo *Q-learning*, a fim de propor um cenário ubíquo ao estudante.

#### 5. TRABALHOS RELACIONADOS

No estudo de [23], é relatado o desenvolvimento de um agente pedagógico animado denominado PAT (um acrônimo para *Pedagogical and Affective Tutor*). O PAT utiliza sensores virtuais para capturar o comportamento do estudante no ambiente virtual, e a partir dessa identificação, representa as emoções: alegria, tristeza, satisfação, frustração, raiva, gratidão e vergonha.

O autor [10] apresenta em seu estudo um agente conversacional inserido no AVA Moodle, que possui o objetivo de auxiliar o estudante na resolução de dúvidas que pudessem surgir no contexto de um curso sobre softwares educacionais. Para isso, o agente realiza a simulação de conversas utilizando linguagem natural com o uso da linguagem AIML. O autor identificou que um agente com capacidade de conversar com usuário é capaz de auxiliar o estudante de cursos realizados em AVA e, ainda, concluiu que um *chatterbot* pode atuar como um tutor pedagógico, contribuindo no processo de aprendizagem ao propiciar um ambiente virtual dinâmico, favorecendo o conteúdo educacional apresentado.

Na pesquisa realizada pelos autores [12], é salientado o uso de um agente inteligente tridimensional que se comunica com o usuário por intermédio de questionamentos em um AVA. De maneira semelhante ao estudo de [23], o agente Dóris é capaz de expressar algumas emoções ao considerar o que o usuário está fazendo no ambiente, sendo elas: tristeza, alegria, surpresa, indignação, expectativa, atenção e dúvida. Em contraponto ao estudo de [10], que discorre sobre um agente que responde às perguntas

realizadas pelo estudante, a pesquisa de [12] apresenta um agente que questiona o usuário sobre o seu comportamento no ambiente, ao consultar o histórico de navegação do usuário. Desse modo, o agente Dóris identifica o que o usuário está aprendendo e emite avisos motivacionais sobre páginas que o usuário deve acessar para realizar um aprendizado eficiente sobre um determinado tópico do curso.

Para ajudar usuários sobre dúvidas referentes a Epilepsia, [14] prepararam um agente conversacional com técnica de aprendizado de máquina. Ao ser questionado, o agente AGEbot verifica em sua base de conhecimento a resposta para a pergunta, de maneira semelhante ao desenvolvido no trabalho de [10]. Se o agente AGEbot não encontra uma resposta apropriada para ajudar a solucionar a dúvida do usuário, armazena essa pergunta em sua memória como um questionamento não solucionado e emite um aviso ao usuário de que ainda não possui resposta. Os autores utilizaram a técnica de aprendizagem por reforço, para que no momento seguinte o agente possa responder corretamente a pergunta. O aprendizado ocorre através do cruzamento da memória de trabalho do agente para a criação de novas regras de produção.

Os autores [11] uniram a estratégia de *Non Player Character* (NPC) à *chatbots* para desenvolver um agente inteligente em um ambiente imersivo. O agente é denominado ELAI e possui a capacidade de apresentar suas respostas de maneira orientada ao nível de conhecimento do usuário. Para isso, verifica a pontuação do usuário a partir de um questionário composto por dez questões referentes a área de redes de computadores. Desse modo, o agente é capaz de identificar o nível de conhecimento do estudante sobre o assunto, e quando questionado utiliza respostas conforme o contexto detectado. Portanto, o agente ELAI é considerado um agente inteligente sensível ao contexto, com recursos de conversação.

Ao considerar o que foi descrito nessa seção, obtêm-se algumas contribuições dos estudos relacionados.

As principais contribuições da pesquisa de [23] consistem em aspectos teórico-práticos, sendo eles: a possibilidade de um agente capturar informações do usuário de um ambiente virtual e a importância da representação gráfica, a fim de tornar a interação entre agente e estudante mais atraente.

O estudo de [10] destaca diversas contribuições a esta pesquisa, sendo que a principal consiste na integração do agente ao AVA Moodle. Já a contribuição do estudo de [12] consiste na utilização de agentes inteligentes no âmbito de ambientes educacionais com o intuito de motivar o estudante durante o processo de aprendizado.

No estudo de [14] é inserida uma técnica de aprendizagem de máquina em um *chatbot*, sendo esta sua principal influência nesta pesquisa. Os autores [11] contribuíram no sentido de que o agente ELAI utiliza sensibilidade ao contexto ao dialogar com o usuário.

Por fim, pode-se concluir que cada um dos trabalhos correlatos apresenta alguma característica importante para a criação do agente inteligente para auxiliar estudantes em um AVA apresentado neste artigo.

## 6. DESCRIÇÃO DO AGENTE UBIBOT

Essa seção tem como foco delinear o agente Ubibot. O agente será disponibilizado para estudantes matriculados na disciplina de Engenharia de *Software* do ensino superior em Ciência da Computação. Desse modo, o Ubibot será inserido na disciplina registrada no AVA Moodle. A escolha do ambiente Moodle justifica-se devido ao mesmo apresentar código aberto, possuir uma grande quantidade de documentação disponível na Internet e ainda, ser utilizado na instituição de ensino onde a pesquisa está sendo realizada.

No estudo de [10], o agente conversacional foi inserido em algumas páginas do Moodle, as quais apresentavam atividades. Nesta pesquisa, o agente será implementado em um bloco, pois o bloco pode ficar disponível em qualquer página do curso, garantindo o suporte em qualquer lugar no ambiente da disciplina. Diante disso, o Ubibot é implementado em um bloco *HyperText Markup Language* (HTML).

O Ubibot apresenta dois módulos para identificação de contexto, sendo um responsável por detectar o nível de conhecimento prévio do estudante, e o outro responsável por identificar o desempenho do estudante durante a interação com o ambiente. O conhecimento do usuário é uma característica importante a ser considerada em sistemas educacionais [5]. No estudo de [4] é destacada a importância de apresentar o conteúdo direcionado ao usuário considerando o seu conhecimento sobre o assunto em questão, visto que os estudantes de uma mesma disciplina podem não apresentar o mesmo conhecimento.

Para identificar o nível de conhecimento do estudante, esta pesquisa baseia-se no estudo de [4]. Desse modo, o estudante deve responder um questionário que contém dez questões sobre Engenharia de *Software*, que abrangem o conteúdo programático da disciplina. O agente considera a pontuação obtida pelo estudante e apresenta suas respostas considerando o fator delimitador exposto na Tabela 1.

**Tabela 1. Identificação do nível de conhecimento do estudante**

Nível de conhecimento	Fator delimitador
Nível Básico	$0 \leq \text{pontuação} \leq 4$
Nível Intermediário	$4 < \text{pontuação} \leq 7$
Nível Avançado	$7 < \text{pontuação} \leq 10$

O agente atua no ambiente por meio de um conjunto de regras de produção, formadas pela condição SE e pela ação ENTÃO. A condição SE corresponde ao fator delimitador e a ação ENTÃO representa a resposta do agente. Nesse sentido, a atuação do agente é constituída de percepções e reações, de maneira semelhante ao agente Dóris de [12].

Diferentemente do ELAI, que apresenta suas respostas considerando sempre o nível de conhecimento dos alunos [11], o Ubibot considera também o desempenho do aluno durante a disciplina. Para isso, acompanha a nota do estudante e verifica o desempenho do mesmo.

O nível de desempenho do estudante é obtido conforme sua média de notas em uma disciplina [31]. Assim, o estudante pode possuir um desempenho baixo, bom ou excelente. Para agrupar o usuário em uma dessas classificações, [31] considera a média de notas geral do aluno na disciplina.

Conforme [31], o aluno é classificado com baixo desempenho se possuir uma média geral menor que 7 pontos, considerando uma

escala entre 0 a 10. Classifica-se com bom desempenho quando possuir uma média geral entre 7 e 8 pontos. Classifica-se com excelente desempenho o estudante que possuir uma média geral maior que 8.

O Ubibot identifica a média geral das notas obtidas pelo aluno nas atividades realizadas na disciplina. Ao considerar o fato de que a cada semana existem novas atividades avaliativas, o agente identifica o total de pontuação disponível para considerar a porcentagem de aproveitamento que o aluno obteve na disciplina. Portanto, o agente considera o fator delimitador apresentado na Tabela 2.

**Tabela 2. Identificação do nível de desempenho do estudante**

Nível de desempenho	Fator delimitador
Baixo desempenho	$0\% \leq \text{media} < 70\%$
Bom desempenho	$70\% \leq \text{media} \leq 80\%$
Excelente desempenho	$80\% < \text{media} \leq 100\%$

O funcionamento do Ubibot considera dois módulos principais. Inicialmente, o agente identifica o nível de conhecimento prévio do aluno sobre Engenharia de *Software*, apresentando respostas de acordo com este nível de conhecimento.

O agente sempre verifica se existe alguma tarefa avaliativa já avaliada no AVA, para que, assim, consiga detectar o nível de desempenho do estudante. Caso positivo, o agente começa a identificar dinamicamente o nível de desempenho e deixa de considerar o nível de conhecimento prévio. Ressalta-se que o nível de conhecimento é utilizado inicialmente para que o agente não fique desorientando ao responder as perguntas do usuário.

O objetivo do agente é fazer com que o usuário consiga melhorar o desempenho na disciplina. Para isso, ele irá ajudar o estudante. Durante a interação com o usuário, o Ubibot emite um *feedback* apropriado conforme o contexto. Para conhecer as ações que deve tomar no ambiente e responder adequadamente às perguntas do usuário, o agente utiliza o algoritmo *Q-learning*.

O agente acompanha as notas do estudante, pois as mesmas representam o ambiente onde o Ubibot está inserido, o nível de desempenho representará o estado do agente no ambiente, as respostas do agente contemplam as ações que são realizadas pelo agente, tendo em vista que o aluno obtenha as respostas adequadas para concluir as tarefas solicitadas pelo professor da disciplina.

Caso o estudante venha a progredir seu nível de desempenho, o agente obterá um reforço positivo, pois suas ações fizeram com que o estudante obtivesse um melhor desempenho na disciplina. Caso contrário, o reforço será negativo, porque suas ações não obtiveram êxito. Desse modo, o algoritmo *Q-learning* auxilia o agente na identificação de estados e melhores ações a serem tomadas, refletindo na base de conhecimento apropriada ao nível de desempenho do estudante. O Ubibot apresenta três bases de conhecimento, cada uma representando um dos contextos que o agente identifica.

Ao considerar o estudo de [14], um *chatbot* pode deixar de responder alguma questão por não possuir uma resposta adequada para a pergunta. Para resolver essa limitação, os autores utilizaram o histórico de interações entre agente e usuário para criar novos pares conversacionais. No Ubibot também é utilizado o método de aprendizagem adotado no agente AGEbot para a construção de novos pares conversacionais.

Para ser considerado um ambiente de aprendizagem ubíqua, deve ser considerada a adaptação a diferentes tipos de dispositivos, ou seja, características para aprendizagem com mobilidade. Com a execução do *plugin* Bootstrap no Moodle, o agente persiste independente de lugar e dispositivo de acesso, considerando um ambiente computacional com suporte à ubiquidade.

Para a implementação do agente, alguns procedimentos e materiais serão necessários, diante disso, a seguir serão elencados os procedimentos metodológicos que foram adotados.

## 7. PROCEDIMENTOS METODOLÓGICOS

Este artigo faz parte de um trabalho de conclusão de curso em desenvolvimento no curso de Ciência da Computação de uma Instituição de Ensino Superior. A pesquisa consiste em elaborar um cenário virtual ubíquo, a partir do desenvolvimento de um agente sensível ao contexto do usuário. Desse modo, para o desenvolvimento do Ubibot serão necessárias cinco etapas: Etapa I – *Insights* sobre o Trabalho e Aspectos Teóricos; Etapa II – Modelagem; Etapa III – Desenvolvimento; Etapa IV – Testes; Etapa V – Análises dos Resultados e Considerações Finais. Ressalta-se que até o presente momento, foram concluídas as três primeiras etapas. Por esse motivo, neste artigo são descritos apenas alguns experimentos que contemplam as etapas iniciais.

A primeira etapa consiste na contextualização dos aspectos teóricos que envolvem o funcionamento do Ubibot, tais como: AVA, aprendizagem eletrônica móvel e ubíqua, consciência de contexto, agentes inteligentes, aprendizagem de máquina e agentes conversacionais. Essa etapa está representada nas seções, 2, 3, 4, 5 e 6 deste artigo, onde foram identificados alguns recursos tecnológicos que podem ser utilizados no agente inteligente, tais como: Moodle, Bootstrap, Program O, AIML, *Q-learning*.

A segunda etapa busca utilizar a *Unified Modeling Language* (UML) para fazer a modelagem do Ubibot, considerando os aspectos definidos nesta seção do artigo. Também faz parte dessa etapa o desenho da arquitetura do agente e a criação do modelo Entidade-Relacionamento para representar as modificações realizadas no banco de dados do ambiente Moodle.

A terceira etapa consiste na implementação do agente junto ao ambiente Moodle, inicialmente por meio da escolha de uma versão do Moodle e as funcionalidades de sensibilidade ao contexto (nível de conhecimento e nível de desempenho) e adaptação da interface do ambiente para o contexto de dispositivos móveis. A etapa contempla a inserção do Módulo Identificador do Nível de conhecimento prévio de Estudantes (MINE) e o Módulo Identificador do Desempenho de Estudantes (MIDE) no Moodle, que são responsáveis por capturar o contexto do estudante. Junto aos módulos MINE e MIDE foi implementado o agente de aprendizagem por reforço, que permite ao agente executar a base de conhecimento correta, considerando o contexto. As bases de conhecimento são hospedadas no Program O, codificadas na linguagem AIML.

Atualmente, o agente é representado por um personagem animado desenvolvido com auxílio da ferramenta *web Voki*<sup>3</sup>.

A quarta etapa consiste na elaboração e realização de testes a fim de verificar o funcionamento do agente e é dividida em duas fases. A primeira consiste na análise dos *logs* de conversação do Ubibot,

<sup>3</sup> Disponível em <http://www.voki.com/>

ou seja, será analisado o desempenho do agente em responder as perguntas emitidas pelos estudantes. A segunda fase consiste em identificar a usabilidade do agente, fazendo uso do questionário *System Usability Scale* (SUS) de Brooke [32].

A quinta etapa consiste na análise de experimentos com o estudante, onde será identificada a validade computacional do Ubibot. Um meio para isso será a utilização de técnicas estatísticas em variáveis quantitativas do questionário SUS. Esta etapa também tem por objetivo evidenciar possíveis trabalhos futuros para a pesquisa.

Diante do exposto, na próxima seção são apresentados alguns resultados decorrentes de experimentos já realizados visando contemplar a proposta descrita neste artigo, objetivando evidenciar as possibilidades e abrangências do contexto dessa pesquisa.

Para realizar o experimento foi utilizado um computador DexPC com processador *Intel Core i5-2320* 3.00 GHz com 4 núcleos físicos, 4 núcleos virtuais, memória RAM de 4,00 GB, disco rígido de 500 GB e sistema operacional *Microsoft Windows Seven Professional* de 64 bits.

O ambiente Moodle (versão 3.0) foi instalado em servidor local, com auxílio do software Xampp que possui o sistema gerenciador de banco de dados *MySQL* (4.5.1), além do Apache para hospedar o sistema, e oferece suporte a interpretação da linguagem de programação *Hypertext Preprocessor* (PHP - versão 5.5.33). Após a instalação, o processo de modificação na codificação do ambiente teve início. Nas páginas do sistema, utilizaram-se as seguintes linguagens: PHP, *JavaScript*, HTML; e para manipular o banco de dados foi utilizado a *Structured Query Language* (SQL).

## 8. EXPERIMENTOS REALIZADOS

### 8.1 Módulo Identificador do Nível de conhecimento prévio de Estudantes (MINE)

O MINE é um módulo embasado nos instrumentos de investigação de [4] e [11], cujo objetivo é identificar o contexto do estudante da disciplina de Engenharia de *Software* do ambiente Moodle. Trata-se de um questionário constituído por dez questões objetivas, cada uma delas possuindo cinco opções de resposta, em que o estudante deve escolher apenas uma alternativa. Vale salientar que a pontuação máxima obtida pelo estudante é dez pontos e a mínima é zero.

Com relação à origem do questionário, as dez questões do MINE foram extraídas do Exame Nacional de Desempenho dos Estudantes (ENADE)<sup>4</sup> e do Exame Nacional para Ingresso na Pós-Graduação em Computação (POSCOMP)<sup>5</sup>. Os autores [4] descrevem em seu trabalho que o professor que ministra a disciplina deve escolher as questões que irá utilizar para identificar o nível de conhecimento. Assim, no MINE foram selecionadas algumas questões que contemplam o conteúdo programático do curso, atendendo ao Projeto Pedagógico do Curso de Graduação em Ciência da Computação da Instituição de Ensino Superior em que a pesquisa está sendo desenvolvida.

<sup>4</sup> Disponível em <http://portal.inep.gov.br/enade>

<sup>5</sup> Disponível em <http://www.sbc.org.br/educacao/poscomp>

Após o estudante responder o questionário, o sistema realiza o somatório dos números que traduzem as opções de resposta, totalizando uma média. Assim, o sistema utiliza o fator delimitador, evidenciado na Tabela 1 da sexta seção desta pesquisa, para identificar o nível de conhecimento.

A funcionalidade do MINE no Ubibot é identificar e registrar na base de dados do Moodle o nível de conhecimento do estudante. Para realizar o registro, foi adicionada uma tabela na base de dados do Moodle. Desse modo, a tabela *mdl\_mine* é destinada ao armazenamento de informações referentes ao contexto diagnosticado pelo MINE.

Para ter acesso ao MINE, o estudante deverá acessar o ambiente Moodle e informar seu *login* e *password*. Após ter acesso ao sistema, o estudante é direcionado a uma página introdutória sobre o agente Ubibot e na sequência à página do questionário, conforme pode ser visualizado na Figura 1. Assim o contexto do estudante é obtido no primeiro acesso ao ambiente, e o usuário é redirecionado para a página da disciplina que contém o agente Ubibot.

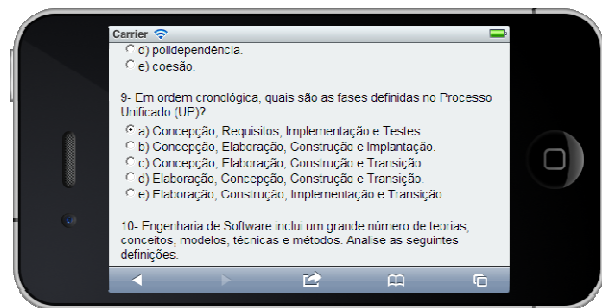


Figura 1. Instrumento utilizado para identificar o nível de conhecimento

### 8.2 Módulo Identificador do Desempenho de Estudantes (MIDE)

O MIDE é um módulo embasado no instrumento de [31], que faz uso de um algoritmo que identifica o desempenho do estudante em atividades realizadas e avaliadas no AVA Moodle. A funcionalidade do MIDE no Ubibot é identificar e registrar na base de dados o nível de desempenho do estudante matriculado na disciplina de Engenharia de *Software*.

O contexto identificado pelo MIDE é realizado de forma dinâmica e invisível para com o usuário, ou seja, este não percebe que o contexto é identificado, pois não necessita responder a um questionário.

### 8.3 Personalização do Moodle ao contexto dos dispositivos móveis

Inicialmente, na seção de plugins do Moodle foi instalado o Bootstrap, e na seleção de temas do ambiente, nas opções Mobile e Tablet, foi escolhido este tema. O tema Aardvark disponível no repositório do Moodle foi escolhido para ser utilizado quando o acesso for realizado através de *desktop* ou *notebook*.

Nessa perspectiva, foram realizados alguns testes, tendo como objeto verificar a funcionalidade do ambiente. Utilizou-se o *desktop* descrito na sétima seção deste artigo, e um dispositivo móvel.

A Figura 2 apresenta o ambiente acessado via dispositivo móvel.



Figura 2. Ubibot sendo acessado por meio de um dispositivo móvel

A Figura 3 apresenta a interface da aplicação, sendo acessada em um *desktop*. Através da mesma é possível identificar a localização do agente inserido na disciplina.



Figura 3. Ubibot disponibilizado na disciplina

## 8.4 Modelagem

No banco de dados do ambiente foram inseridas duas tabelas, conforme descrito no decorrer desta seção, ambas para armazenar os dados de contexto. A tabela *mdl\_mine* é destinada ao armazenamento de informações referentes ao nível de conhecimento prévio do estudante da disciplina de Engenharia de Software e a tabela *mdl\_mide* é responsável por armazenar informações sobre o desempenho do estudante de forma dinâmica durante sua interação.

Primeiramente, o usuário deve acessar o Moodle, que identifica seu papel no ambiente, caso o papel do usuário seja “estudante”, o Ubibot executa o algoritmo *Q-learning* para identificar o contexto.

O agente, com auxílio do *Q-learning*, inicialmente verifica se o contexto de desempenho do estudante já foi registrado, identificando se existe alguma tarefa já avaliada na disciplina. Caso positivo, o MIDE identifica o nível de desempenho do estudante e o Ubibot carrega a base de conhecimento correta ao respectivo contexto. Caso não exista tarefa avaliada, o agente executa o MINE.

O MINE verifica se o nível de conhecimento do estudante já foi identificado anteriormente, e caso positivo o Ubibot carrega a base de conhecimento correta; caso o nível de conhecimento ainda não tenha sido detectado, o ambiente apresentará uma mensagem de apresentação do Ubibot e em seguida, o questionário MINE é disponibilizado ao usuário. Após o estudante responder o questionário, o agente fica ciente da base de conhecimento que deverá utilizar quando o usuário lhe fizer questionamentos.

A Figura 4 apresenta os fluxos de controle do Ubibot.

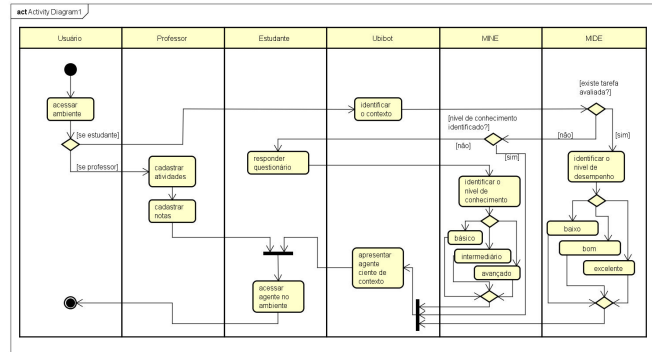


Figura 4. Diagrama UML de atividades: fluxos de controle do Ubibot

## 8.5 Avatar de interface

Conforme [23], a existência de uma representação gráfica de um agente no ambiente virtual tornar-se necessária, pois um avatar pode dar a impressão ao estudante que o mesmo não está sozinho no ambiente. Assim, para representar o Ubibot fez-se uso da ferramenta *online* Voki, que permite criar e personalizar avatares de duas dimensões.

O avatar que representa o Ubibot é desenvolvido em Flash Player, pois é a tecnologia que a Voki utiliza. Por este motivo, o dispositivo que executa o agente deve conter o plugin Flash Player habilitado no *browser*. Os dispositivos móveis que possuem o sistema operacional Android não contêm o Flash Player, visto que nos últimos anos com o surgimento de novas tecnologias como HTML5, o Flash Player vem sendo substituído. O uso da tecnologia Voki tornou-se uma limitação do agente. Nesse sentido, para a implementação efetiva do Ubibot será necessário criar um avatar com outra tecnologia, como o HTML5.

## 8.6 Implementação da base de conhecimento

A base de conhecimento do agente é construída na linguagem de marcação AIML, específica para agentes que têm o intuito de se comunicar por meio de linguagem natural. Ao considerar o fato que o agente irá auxiliar o estudante de maneira orientada ao contexto, criou-se três bases de conhecimento. A primeira base de conhecimento é destinada aos estudantes que possuem o nível de conhecimento básico ou um baixo desempenho na disciplina. A segunda é designada aos alunos que apresentam conhecimento intermediário ou um bom desempenho. Por fim, a terceira base é voltada aos discentes que possuem conhecimento avançado ou um excelente desempenho.

A Figura 5 apresenta um exemplo, sobre a definição de engenharia de *software*. Nela é possível perceber que o padrão de entrada *<pattern>* é “o que é engenharia de *software*” e o texto resposta será o que está dentro da *tag template*. Esse exemplo pertence à base de contexto básico.

```

1. <aiml>
2. <category>
3. <pattern>o que é engenharia de software
</pattern>
4. <template>
5. É uma disciplina que estuda todos os
procedimentos de produção de um software. Desse
modo, está relacionada com a prática de
desenvolvimento e entrega de software útil.
6. </template>
7. </category>
8. </aiml>

```

**Figura 5. Código em AIML pertencente a base de conhecimento de contexto básico**

Caso o estudante realizar a mesma pergunta e estiver na categoria de contexto intermediário, o mesmo receberá outra resposta, considerada adequada ao seu contexto. Na Figura 6, é possível observar um exemplo de resposta <template> tratada para o contexto intermediário.

```

1. <aiml>
2. <category>
3. <pattern> o que é engenharia de software
</pattern>
4. <template>
5. Engenharia de Software é uma área da
ciência da computação voltada à especificação,
desenvolvimento e manutenção de sistemas de
software, com aplicação de tecnologias e práticas
de gerência de projetos e outras disciplinas,
visando organização, produtividade e qualidade.
6. </template>
7. </category>
8. </aiml>

```

**Figura 6. Código em AIML pertencente a base de conhecimento de contexto intermediário**

A Figura 7 apresenta um exemplo em que a resposta é realizada pelo agente, considerando a base de conhecimento de contexto avançado.

```

1. <aiml>
2. <category>
3. <pattern> o que é engenharia de software
</pattern>
4. <template>
5. É uma área de estudo que abrange um
conjunto de três elementos fundamentais:
métodos, ferramentas e procedimentos que
possibilita ao gerente o controle do processo de
desenvolvimento de software e oferece ao
profissional uma base para a construção de
software de alta qualidade. Envolve um amplo
conjunto de tarefas que incluem planejamento e
estimativas de projeto, análise de requisitos de
software e de sistemas, projeto estruturado de
dados, especificação, codificação, teste e
manutenção.
6. </template>
7. </category>
8. </aiml>

```

**Figura 7. Código em AIML pertencente a base de conhecimento de contexto avançado**

## 9. CONSIDERAÇÕES FINAIS

Este artigo teve como objetivo evidenciar a possibilidade de criar agentes inteligentes para ambiente virtuais de aprendizagem, com o objetivo de oferecer aspectos de aprendizagem ubíqua ao estudante. Para isso partiu-se da proposta de criar um agente com capacidade de auxiliar o estudante de Engenharia de Software, considerando inicialmente o nível de conhecimento e posteriormente, de forma dinâmica e não intrusiva, o nível de desempenho na disciplina.

O artigo apresenta três de cinco etapas da pesquisa de um trabalho de conclusão de curso em realização pelo autor principal deste estudo. Elucidou-se as etapas de desenvolvimento do agente Ubibot e a sua modelagem realizada em UML. Também demonstrou-se a aplicabilidade do agente, por meio de demonstração do seu funcionamento durante a implementação da proposta.

Identificou-se que a utilização da ferramenta Voki não é adequada a esta pesquisa, pois os avatares criados com tecnologia Flash Player não executam em dispositivos móveis com sistema operacional Android, impedindo a característica de mobilidade, essencial para a aprendizagem ubíqua.

O próximo passo da pesquisa será identificar uma nova ferramenta para criação do avatar. Após isto, será realizada a aplicação do agente Ubibot com estudantes da disciplina de Engenharia de Software. Durante a aplicação, o agente utilizará o método de aprendizado de máquina proposto por [14], que objetiva identificar se está respondendo corretamente às perguntas do estudante.

A avaliação será realizada durante um período de dois meses no contexto da disciplina. Acredita-se que com esse tempo será possível identificar as principais contribuições do agente, assim como as limitações da pesquisa. Ao final do período em que o Ubibot estará disponível aos alunos, os mesmo realizarão a avaliação de usabilidade do agente, por meio de um instrumento SUS, que pretende evidenciar alguns aspectos relevantes, como: complexidade do agente, facilidade na utilização e necessidade de suporte para utilização.

## 10. REFERÊNCIAS

- [1] P. M. Mozzaquatro, et al., "Mobile Learning Engine Moodle adaptado aos diferentes estilos cognitivos utilizando Hipermedia Adaptativa," in *Conference IADIS Ibero-Americana WWW/Internet 2010* (CIAWI 2010), 2010.
- [2] F. B. Nunes, et al., "An adaptive environment based on Moodle with treating of quality of context," *International Journal of Knowledge and Learning* (Print), v. 10, p. 198-221, 2015.
- [3] I. Gasparini, et al., "Modelo do usuário sensível ao contexto cultural em um sistema e-learning adaptativo," *Informática na Educação: teoria & prática*, v. 14, n. 1, p. 123-135, jun, 2011.
- [4] C. C. Possobom, et al., "SistEX Um sistema dinâmico para detectar a experiência do aluno," in *Proc. Moodle Moot Brasil*, 2014, p.98-108.
- [5] N. S. R. S. Santos, et al., "Monitoramento navegacional do aluno para descoberta de padrões de preferências de aprendizagem no Moodle," *Cadernos de Informática*, v. 6, n.1, jun, 2011.
- [6] S. D. Piovesan, et al., "U-SEA: A Learning Environment Ubiquitous Using Cloud Computing," *International Journal: Emerging Technologies in Learning*, v. 7, p. 19-23, 2012.
- [7] P. M. Mozzaquatro, et al., "Modelagem de um framework para adaptação de ambientes virtuais de aprendizagem móveis aos diferentes estilos cognitivos," *Revista Novas Tecnologias na Educação (RENTE)*, v.7, n. 3, p.01-11, dez, 2009.
- [8] S. Yahya, et al., "The definition and characteristics of ubiquitous learning: A discussion," *International Journal of*

- Education and Development using Information and Communication Technology*, v. 6, p. 117–127, 2010.
- [9] D. N. F. Barbosa, *et al.*, “Learning in a Large-Scale Pervasive Environment,” in *Proc. 4th IEEE Conference on Pervasive Computing and Communications Workshops (PERCOM)*, 2006, pp. 226-230.
- [10] L. M. A. Santos, “A Inserção de um Agente Conversacional Animado em um Ambiente Virtual de Aprendizagem a partir da Teoria da Carga Cognitiva,” Tese, Centro Interdisciplinar de Novas Tecnologias na Educação, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil, 2009.
- [11] F. Herpich, *et al.*, Three-Dimensional Virtual Environment and NPC: A Perspective about Intelligent Agents Ubiquitous. In: Francisco Milton Mendes Neto; Rafael de Souza; Alex Sandro Gomes. (Org.). Handbook of Research on 3-D Virtual Environments and Hypermedia for Ubiquitous Learning. 1ed.: IGI Global, 2016, v. 21, p. 510-536. DOI: 10.4018/978-1-5225-0125-1.ch021
- [12] A. K. Silva, *et al.*, “O método clínico de Piaget como forma de avaliação da aprendizagem em um sistema tutor inteligente com agente pedagógico,” in *Proc. of the Simpósio Brasileiro de Informática na Educação*, 2010.
- [13] E. M. Bada and C. S. Menezes, “Uma proposta para extração de perguntas e respostas,” in *Proc. of the XVII Congreso Internacional de Informática Educativa (TISE)*, 2012, pp. 44-49.
- [14] M. C. Fossatti, *et al.*, “O uso de um chatterbot em AIML como módulo de interface de um Sistema Tutor Inteligente para auxiliar no aprendizado de Epilepsia,” in *Proc. of the XXII Simpósio Brasileiro de Informática na Educação*, 2011, pp. 164-167.
- [15] R. M. Andrade, “Mobile bot: um chatterbot educacional para dispositivos móveis,” *Revista Brasileira de Computação Aplicada*, v.4, n.2, p. 83-91, out, 2012.
- [16] F. P. Galafassi, *et al.*, “Análise Crítica das Pesquisas Recentes sobre as Tecnologias de Objetos de Aprendizagem e Ambientes Virtuais de Aprendizagem,” *Revista Brasileira de Informática na Educação*, v.21, n.1, jun, 2013.
- [17] P. C. Oliveira, *et al.*, “Ambientes Virtuais de Aprendizagem: Revisão Integrativa de Teses de Doutorado no Contexto Brasileiro entre 2003 e 2012,” *Revista Novas Tecnologias na Educação (RENOTE)*, v. 12, n.2, dez, 2014.
- [18] G. M. Machado, *et al.*, “Uma revisão sistemática sobre as abordagens ubíquas para recomendação educacional: estariam elas se tornando adaptativas?,” in *Proc. XXVI Simpósio Brasileiro de Informática na Educação*, 2015, pp. 170-179.
- [19] M. Weiser, “The Computer for the 21st Century,” *Scientific American*, v. 265, n. 3, p. 94–104, sep 1991.
- [20] L.S. Vygotsky, “Mind in Society,” *Harvard University Press*, 1978.
- [21] S.J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2010.
- [22] S. Franklin and A. Graesser, “Is it an Agent, or just a Program?: a taxonomy for autonomous agents,” in *Proc. of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages*, v.3, p.21–35, 1996.
- [23] P. A. Jaques and R.M. Vicari, “PAT: um agente pedagógico animado para interagir afetivamente com o aluno,” *Revista Novas Tecnologias na Educação (RENOTE)*, v. 3, n.1, jun, 2005.
- [24] M. D. Leonhardt, “Doroty: um Chatterbot para Treinamento de Profissionais Atuantes no Gerenciamento de Redes de Computadores,” Dissertação, Instituto de Informática, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil, 2005.
- [25] F. S. Sgobbi, *et al.*, “A utilização de agentes inteligentes no apoio ao autocuidado de idosos,” *Revista Novas Tecnologias na Educação (RENOTE)*, v. 12, n.2, dez, 2014.
- [26] M. Thiry and D. E. Rios, “Um chatterbot para auxiliar no aprendizado em qualidade de software,” in *Proc. of the XI Congreso Argentino de Ciencias de la Computación*, 2005, pp. 01 –11.
- [27] L. L. Guimarães, “Inferência de Contexto para Dispositivos Móveis utilizando Aprendizagem por Reforço” Dissertação, Universidade Federal do Amazonas, Amazonas, Brasil, 2015.
- [28] C. J. C. H. Watkins, “Learning from Delayed Rewards,” Cambridge University, England, 1989.
- [29] F. A. Dorça, Uma abordagem Estocástica Baseada em Aprendizagem por Reforço para Modelagem Automática e Dinâmica de Estilos de Aprendizagem de Estudantes em Sistemas Adaptativos e Inteligentes para Educação a Distância,” Tese, Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia, Minas Gerais, Brasil, 2012.
- [30] M. V. C. Guelpeli, “Utilização de Aprendizado por Reforço para Modelagem Autônoma do Aprendiz em Sistemas Tutores Inteligentes,” Dissertação, Instituto Tecnológico de Aeronáutica, Brasil, 2003.
- [31] A. F. M. Dias, *et al.*, “Fatores determinants da retenção estudantil em um curso de graduação em engenharia de produção,” in *Proc. of the Congresso Brasileiro de Educação em Engenharia*, 2009, pp. 01- 10.
- [32] J. Brooke, “SUS- A quick and dirty usability scale,” *Usability evaluation in industry* v. 189, p. 194, 1996.