

DESENVOLVIMENTO DE UMA ONTOLOGIA SOBRE PLANOS DE AULA DE FÍSICA COM O USO DA ROBÓTICA EDUCACIONAL PARA APLICAR AO SISTEMA SCIENCE.

João Tadayoshi Tatsukawa Neto¹, Silvio Roberto Fernandes de Araújo², Paulo Gabriel Gadelha Queiroz³

Resumo: A utilização de metodologias de ensino é essencial para uma construção eficiente do conhecimento dos alunos em sala de aula. Na disciplina de Física se faz necessário a utilização de métodos de ensino dinâmicos e eficientes, pois é notável uma dificuldade por parte dos estudantes do ensino médio para entender os assuntos, principalmente por não conseguirem visualizar os fenômenos acontecendo na realidade. Sendo assim, a robótica é uma ferramenta didática que tem potencial de proporcionar dinamismo para a aula, fazendo com que os alunos aprendam de maneira lúdica. Apesar da robótica ser uma ferramenta interessante a ser aplicada, o fato de os professores não saberem manipular o material dificulta a sua utilização. Tendo em vista esse problema, foi desenvolvido o Sistema de Geração Integrado de planos de aula para o ENSINO apoiado por Robótica Educacional (SCIENCE), que é uma plataforma que fornece aos professores planos de aula de matemática com o uso de robótica educacional. A fim de contribuir e expandir as áreas de conhecimento da plataforma SCIENCE, foram implementados, nesse trabalho, planos de aula de física com o uso de Robótica Educacional que foram disponibilizados por meio de uma API REST para futura integração ao sistema. Para se obter as informações necessárias para a criação de um plano de aula foi desenvolvido uma ontologia de domínio, construída a partir da metodologia *Ontology Development 101*, e verificada a partir da linguagem de consulta SPARQL.

Palavras-chave: Ontologia; Robótica Educacional; Física; LEGO; API

1. INTRODUÇÃO

A educação exerce um papel fundamental na sociedade, na qual o indivíduo se torna capaz de entender um determinado problema e levantar uma possível solução, além de questionar situações ao seu redor. Assim sendo, o ser humano está sempre realizando alterações no processo de aprendizagem de maneira que ela se adeque às necessidades do mundo atual [29].

Zilli [29] expõe que através do uso de novas tecnologias é possível estimular o aluno a desenvolver novas formas de aprendizado por meio da busca por conhecimento e novos estilos de raciocínio.

É perceptível que a dificuldade no processo de aprendizado da disciplina de física é um problema que atinge uma grande parcela dos alunos. Barroso, Rubini e Silva [2] apresentam indícios a partir de dados referentes ao desempenho de estudantes em algumas questões de física no ENEM, que há dificuldade na compreensão dos conceitos básicos em diversos assuntos da disciplina por parte da maioria dos concluintes do ensino médio, dessa forma se faz necessário o uso de novas metodologias de ensino para explicar os conceitos de maneira mais clara e dinâmica.

Para Passos [18] tendo em vista as inovações na área de ensino, de maneira a alcançar uma melhor didática aos alunos, a robótica é uma ferramenta que pode contribuir como um recurso para a interdisciplinaridade, por envolver diversos campos, como a matemática, física, informática etc. Zilli [29], afirma que a robótica educacional tem capacidade de desenvolver nos alunos diversas competências como raciocínio lógico, capacidade crítica e resolução de problemas por meio de erros e acertos.

A Robótica Educacional (RE) tem a capacidade de facilitar o entendimento de assuntos complexos e instigar a curiosidade nos alunos para o conhecimento sobre física devido às experiências em situações reais que ela proporciona. Dessa forma, Santos e Menezes [25] afirmam que:

¹ Graduando do curso Interdisciplinar em Ciência e Tecnologia. Departamento de Ciências Exatas e Naturais – UFERSA. E-mail: jn.tatsukawa@gmail.com

² Orientador, Prof. Dr. Departamento de Computação – UFERSA. E-mail: silvio@ufersa.edu.br

³ Coorientador, Prof. Dr. Departamento de Computação – UFERSA. E-mail: pgabriel@ufersa.edu.br

O processo de construção do conhecimento físico é um processo histórico, produzido em sociedade, associado com outras formas de expressão e produção humanas. É importante reconhecer, portanto, que o conhecimento da Física “em si mesmo” não é o objetivo final, mas deve ser entendido como um meio, um instrumento para compreensão do mundo, podendo ser prático mas, permitindo ultrapassar o interesse imediato, produzindo no aluno a sua identidade de indivíduo criativo, social e possuidor de atitudes, hábitos e habilidades úteis a si mesmo e à sociedade.

Uma das metodologias que pode despertar maior interesse nos alunos é o ensino com o apoio de Robótica Educacional. Contudo, os projetos de extensão PENSARE⁴ e Robot em Ação⁵, executados pela UFERSA identificaram que uma grande parcela dos professores não é devidamente capacitada para utilizar a robótica em sala de aula, assim surge o problema do trabalho: “Como ajudar os professores do ensino médio a fazer o uso da robótica educacional como ferramenta de aprendizagem para aulas de física de maneira prática?”

Embora a robótica seja um recurso didático que pode agregar às aulas, o fato de professores não possuírem o domínio sobre a ferramenta, dificulta o uso da mesma. Então, Passos [22], desenvolveu o Sistema de Geração Integrado de planos de aula para o ENSINO apoiado por Robótica Educacional (SCIENCE), como sendo uma plataforma de geração de plano para aulas de matemática para o Ensino Fundamental II, que utiliza uma ontologia para descrever os conceitos sobre Robótica Educacional aplicado as aulas de matemática.

Esse trabalho tem por objetivo principal auxiliar os professores de física a utilizar a robótica como ferramenta didática durante as aulas, instruindo-os a utilizar a ferramenta, para promover aos alunos um melhor entendimento acerca do assunto abordado. Para isso foram definidos dois objetivos específicos sendo eles: desenvolver uma *Application Programming Interface* (API) que será integrada ao sistema SCIENCE [22], responsável por fornecer planos de aula de física com a utilização de robótica educacional; e construir uma ontologia com atividades e programações relacionando robótica à disciplina de física e a robótica educacional.

O restante desse trabalho foi organizado da seguinte forma. Na seção 2, serão apresentadas as definições de Robótica Educacional e ontologia. Na seção 3, serão apresentados os procedimentos e métodos utilizados para a construção da ontologia e API REST. Na seção 4, serão apresentados os resultados das atividades desenvolvidas no trabalho. E na seção 5, serão apresentadas as considerações finais.

2. FUNDAMENTAÇÃO TEÓRICA

Nesta seção será abordado o conceito de Robótica Educacional e ontologia, que são dois fundamentos necessários para a obtenção do objetivo do trabalho. O primeiro para entender como a robótica pode contribuir para o ensino de Física e o segundo como foi construído a estrutura de informações presentes na API.

2.1. Robótica Educacional

Visando auxiliar a sociedade com tarefas complexas ou repetitivas, o ser humano está sempre em constante evolução na busca de ferramentas que possam contribuir para melhorar o meio onde vive. A robótica é uma área que surgiu para sanar essa necessidade, devido a otimização, custo e praticidade que ela traz. Ela está sempre em expansão sendo considerada uma área interdisciplinar por abordar diversos conceitos físicos, matemáticos e computacionais em suas aplicações. É possível encontra-la em situações diversas no cotidiano, como por exemplo, em casas, hospitais, veículos e na educação [28].

Segundo Campos [4], a Robótica surge como um recurso tecnológico de aprendizagem na educação, na qual ela proporciona a interdisciplinaridade de maneira lúdica, despertando a curiosidade e o interesse dos alunos por meio da vivência de situações reais com os assuntos de sala de aula. Garcia [9] e Zilli [29], demonstram que a Robótica Educacional tem o potencial de gerar muitos benefícios para os estudantes, como por exemplo:

- Desenvolvimento de raciocínio lógico de programação;
- Capacidade de resolução de problemas do cotidiano;
- Senso crítico na conclusão da atividade;
- Emprego da criatividade;
- Interpretação de dados;
- Desenvolvimento de relações interpessoais e intrapessoais;
- Estimulação de aplicação de teorias formuladas;
- Desenvolvimento da concentração e observação.

Na atualidade a robótica está cada vez mais ganhando espaço no meio acadêmico, na qual está se tornando comum a utilização de kits de Robótica, que visam facilitar o manuseio, agregando um contexto lúdico na sala de aula. Um dos exemplares mais utilizados são os kits de modelo da LEGO da linha Mindstorms [18], onde neles é possível encontrar motores, eixos, polias, sensores, engrenagens, conectores etc. Além do material, a empresa

⁴ Disponível em: <https://pensare.ufersa.edu.br/>

⁵ Disponível em: <https://robotemacao.ufersa.edu.br/>

também fornece um *software* próprio para programar o robô, o EV3 Classroom⁶, que apresenta comandos simples e intuitivos no formato de blocos que conectam entre si, formando uma pilha de programação.

Segundo Santos e Menezes [25], a proposta dos kits da LEGO de Robótica Educacional é apresentar conceitos físicos de maneira que a compreensão dos assuntos seja mais clara, vivenciando experiências cotidianas. Sendo assim eles são utilizados como instrumentos didáticos, oferecendo aos alunos uma aula mais dinâmica e participativa, desenvolvendo uma aprendizagem em grupo.

A LEGO na atualidade conta com quatro modelos de kits de robótica, que são: Lego RCX, Bloco NXT 2.0, Bloco EV3 e o Conjunto SPIKE. O Lego Mindstorms EV3 é o mais utilizado e difundido dentre esses modelos, devido ao tempo de mercado e atualizações [18].

O Lego Mindstorms EV3 caracteriza-se por ter 64MB de RAM, Sistema Operacional Linux e entrada para cartão Micro SD. Na Figura 1, é demonstrado esse kit que apresenta os seguintes componentes: 2 motores grandes, 1 motor médio, 1 sensor de cor, 1 sensor ultrassônico, 2 sensores de toque, 1 sensor de giro, 1 bateria, 1 bloco EV3 e peças de encaixe lego para a montagem da estrutura do robô [18].



Figura 1. Kit LEGO Mindstorms EV3. [16]

2.2. Ontologia

A ontologia pode ser descrita como a especificação explícita de uma conceituação. Segundo Gruber [12], para a filosofia esse termo é um relato sistemático da existência, porém no âmbito computacional, o conceito é modificado, pois a existência se torna aquilo que é representado, um determinado conhecimento que é caracterizado como o domínio, sendo ele ponto principal da ontologia. Esse domínio é relacionado com um conjunto de objetos que se relacionam para formar o conhecimento.

As ontologias podem ser utilizadas para implementar e especificar qualquer sistema complexo de computação, podendo ser usadas de diversas maneiras. Dessa forma, Duarte e Falbo [7], expõem que de modo geral ela é utilizada para alcançar três objetivos gerais, sendo eles:

- Auxiliar na compreensão de uma determinada área de conhecimento, de modo que seja explicado o domínio de análise;
- Auxiliar a atingir um consenso sobre determinada área de conhecimento, de maneira que ocorra uma unificação do entendimento dos conceitos envolvidos;
- Auxiliar no aprendizado de certa área de conhecimento, para que assim não seja necessário o indivíduo recorrer a um especialista na área.

Gruber [12] afirma que a ontologia é definida como um conjunto de termos representacionais, onde os conceitos se relacionam com o domínio por meio de alguns componentes. Dessa forma Noy e McGuinness [21], demonstram que há quatro componentes que constituem uma ontologia que são: as classes, funções, instâncias e relações.

2.2.1. Ferramentas para construir uma ontologia

Para auxiliar a construção e manipulação de uma ontologia, diversas ferramentas podem ser utilizadas. Carlan [5] cita quatro ferramentas que podem ser utilizadas, que são elas: Ontolingua; JOE (Java Ontology Editor); OntoEdit; e Protégé. Utilizado na construção da ontologia desse artigo, o Protégé é um *software* desenvolvido por Mark Musen, que conta com diversas funcionalidades, permitindo que as etapas estabelecidas na metodologia sejam satisfeitas [21].

Ao utilizar essas ferramentas, há a necessidade da utilização de um mecanismo de inferência, para verificar e validar se os termos e relações presentes na ontologia estão descritos corretamente, Passos [22] destaca a existência dos seguintes motores de inferência: FaCT ++, Pellet e HermiT. Nesse trabalho foi utilizado o HermiT como mecanismo de inferência.

⁶ Disponível em: <https://bityli.com/xkMwWa>

2.2.2. Linguagem de consulta SPARQL

De acordo com Segundo [26], o SPARQL é um pacote de especificações que fornecem linguagens e protocolos para consultar e manipular o conteúdo publicado em RDF. Passos [22] expõe que para realizar consultas de uma ontologia, o SPARQL é a linguagem de consulta padrão recomendada pelo W3C, ela é baseada em padrões de grafos, fundamentada em triplas RDF, sendo composta por sujeito, predicado e objeto.

Para a consulta em uma ontologia, a estrutura padrão pode ser observada na Figura 2, na qual ela é dividida em três etapas: a declaração de prefixo, que é utilizada para abreviar as URIs; a cláusula de resultado, para declarar as informações que a consulta deve retornar; e o padrão de consulta, para especificar as relações e o que consultar dentro da ontologia [26].

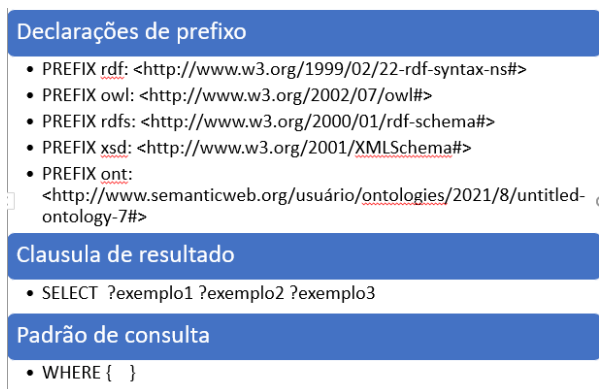


Figura 2. Estrutura da linguagem de consulta SPARQL (Autoria própria)

3. PROCEDIMENTOS E MÉTODOS

Nesta seção são descritos os métodos utilizados para atingir o objetivo. Para isso a realização do trabalho ocorreu em quatro etapas, sendo a primeira o estudo da plataforma SCIENCE e a ontologia desenvolvida por Passos [22]. A segunda etapa, a construção dos planos de aulas de física com o uso da Robótica Educacional. A terceira, o desenvolvimento da ontologia. E a quarta etapa, a criação da API REST que poderá ser implementada ao sistema SCIENCE.

3.1. Estudo do sistema SCIENCE

Primeiramente foi necessário realizar o estudo do desenvolvimento do sistema SCIENCE e como foram construídos os planos de aula, visando ampliar o sistema SCIENCE para planos de aula de física utilizando robótica educacional.

O sistema SCIENCE foi desenvolvido com o intuito de auxiliar professores da disciplina de matemática a utilizar a Robótica Educacional como recurso didático, para isso foi construído por Passos [22], uma ontologia denominada de Ontologia de **ENSINO** e **Aprendizagem** de **Robótica Educacional** (Onto-ENSINARE), nela o domínio definido foi: conteúdos de matemática do Ensino Fundamental II auxiliados pela robótica educacional. O autor utilizou a linguagem de consulta SPARQL, na qual foram definidas questões de competência que a ontologia deve responder corretamente, visando verificar se a mesma está consistente.

Para formar um plano de aula completo, Passos [22], definiu que os seguintes atributos devem estar presentes: Conteúdo, Assunto, Referência, Disciplina, Série, Duração, Recurso Didático, Objetivo, Atividade, Programação e Montagem do robô.

Com relação a estruturação dos componentes do sistema SCIENCE, foi feito o uso do padrão arquitetural MVC (Model – View – Controller), dividindo a aplicação em três camadas: visão, modelo e controle. Na Figura 3 é demonstrado o diagrama de componentes desse sistema.

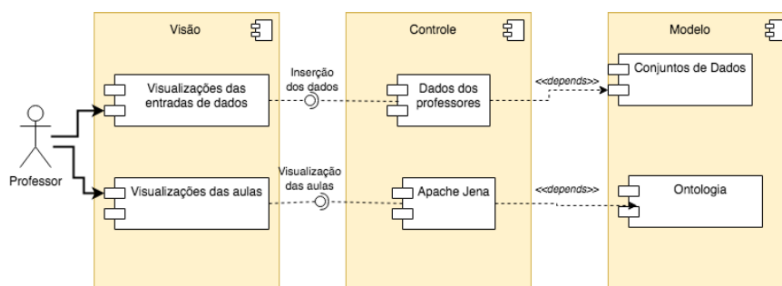


Figura 3. Diagrama de componentes do sistema SCIENCE. [22]

A ontologia e a API REST criadas nesse trabalho foram construídas visando padronizar com as aulas já existentes no sistema, portanto foi utilizado a mesma metodologia seguida por Passos [22] para o desenvolvimento dos planos de aula e construção da ontologia.

3.2. Formação dos planos de aula de Física

Para fornecer a ontologia planos de aula de física com o uso de Robótica Educacional primeiro foi utilizado a Base Nacional Curricular Comum (BNCC) [3], que é um documento que define todos os assuntos que os alunos das escolas brasileiras devem aprender, na qual foi consultado as competências e habilidades que devem ser desenvolvidas na disciplina de Física. A construção dos planos de aula foi feita a partir do livro de física “Conexões com a Física” [10], que retrata de maneira didática os assuntos escolhidos.

Os planos de aula produzidos apresentam os seguintes componentes: série; duração; conteúdo; assunto; programação; recursos didáticos; e atividade. Eles seguem um mesmo padrão de organização, que é dividido nas seguintes etapas: (1) Divisão da turma em equipes; (2) Apresentação da aula e explicação do conteúdo; (3) Montagem do Robô; (4) Explicação dos blocos de programação; (5) Realização das atividades; (6) Desmontar o robô. A seguir serão expostos os planos de aula desenvolvidos.

Os planos de aula foram desenvolvidos visando a utilização do kit Lego Mindstorms EV3, com o *software* EV3 Classroom desenvolvido pela empresa LEGO.

3.2.1. Velocidade Média

Série: 1º ano

Duração: 2 Horas

Conteúdo: Cinemática

Recursos didáticos: Pincel, Quadro, Régua, Data Show, Computador e Kit da Lego EV3 Mindstorms.

Atividade:

1- Em um espaço aberto marque a distância de 100 cm com uma régua ou trena. Em seguida programe o robô para percorrer em linha reta o espaço delimitado com a potência dos motores de locomoção em 25%.

2- Com o auxílio do bloco “temporizador”, programe o robô para exibir na tela o tempo que levou para percorrer o trajeto, em seguida calcule a velocidade em metros por segundos e anote o resultado.

3- Repita o processo alterando a potência dos motores de locomoção para 50, 75 e 100%.

4- Com base nos valores obtidos, qual a relação entre o tempo gasto em cada caso com os resultados das velocidades?

3.2.2. Aceleração

Série: 1º ano

Duração: 2 Horas

Conteúdo: Cinemática

Assunto: Aceleração

Recursos didáticos: Pincel, Quadro, Data Show, Computador e Kit da Lego EV3 Mindstorms.

Atividade:

1- Com base no conteúdo visto em sala de aula, construa com que o robô acelere utilizando apenas comandos de ir para frente. Altere a força dos motores de 5 em 5, começando de 0 até 100. Após isso explique o que ocorreu e o que pode ser melhorado na programação.

2- Refine sua programação utilizando blocos do menu “controle” e “operadores”, para corrigir os erros da questão 1.

3.2.3. Conversão de rotações para centímetros

Série: 1º ano

Duração: 2 Horas

Conteúdo: Sistema internacional de unidades

Assunto: Conversão de rotações para centímetros

Recursos didáticos: Pincel, Quadro, Data Show, Computador e Kit da Lego EV3 Mindstorms.

Atividade:

1- Marque em uma superfície plana a distância de 20cm com uma régua, em seguida programe o robô para percorrer o trajeto utilizando um bloco de movimento com duração em rotações.

2- Realize uma programação que converta centímetros para rotações, em seguida execute a tarefa anterior e verifique se a distância está correta. Para isso utilize a seguinte equação: $\text{Rotações} = \frac{\text{distância em centímetros}}{\pi \cdot \text{diâmetro da roda}}$

3- Com base na programação realizada no exercício anterior faça o robô locomover-se formando um quadrado de 10 cm².

3.2.4. Intensidade da luz refletida

Série: 2º ano

Duração: 2 Horas

Conteúdo: Reflexão da luz

Assunto: Intensidade da luz

Recursos didáticos: Pincel, Quadro, Data Show, Computador, Kit da Lego EV3 Mindstorms e fitas com as cores azul, verde, branco e preto.

Atividade:

1- Com o sensor de cor acoplado do robô, faça uma programação que ao apertar o botão do meio do EV3, a leitura da intensidade de luz refletida nas fitas de cores azul, verde, branco e preto seja feita e exibida na tela. Realize a leitura separadamente de cada fita, anote o resultado e explique o significado físico da variação da intensidade em cada cor.

2- Em uma superfície plana coloque as fitas azul, verde e preto espaçadas a uma distância de 30cm. Realize um programa que aumente a velocidade do robô em 20% a cada cor diferente captada.

3.2.5. Ondas unidimensionais

Série: 2º ano

Duração: 2 Horas

Conteúdo: Fenômenos ondulatórios

Assunto: Ondas unidimensionais

Recursos didáticos: Pincel, Quadro, Data Show, Computador, Kit da Lego EV3 Mindstorms e papel.

Atividade:

1- Realize uma programação que imprima no papel uma onda unidimensional, após isso calcule o período, frequência e velocidade de propagação.

2- Realize o mesmo procedimento aumentando a amplitude da onda, caso os dados sejam alterados, explique o motivo.

3.2.6. Lançamento horizontal

Série: 1º ano

Duração: 2 Horas

Conteúdo: Cinemática

Assunto: Lançamento horizontal

Recursos didáticos: Pincel, Quadro, Data Show, Computador, Kit da Lego EV3 Mindstorms, régua e bolinha de ping-pong.

Atividade:

1) Com o robô execute uma programação para seja arremessado uma bolinha de ping-pong a uma velocidade de 100% a uma altura "h" do chão escolhida durante a atividade.

2) Realize a medição da altura h e do tempo de queda após o início do experimento, em seguida calcule a distância que a bolinha percorreu e a velocidade ao atingir o solo. Considere $g = 10 \text{ m/s}^2$

3.3. Construção da Ontologia

Desenvolvido por Passos [22], o SCIENCE é um sistema de geração de planos de aula de matemática, que apresenta em seus planos os seguintes itens: componentes curriculares, assunto, conteúdo, recurso didático, objetivo, atividades, robôs, montagens e programação. Para essa geração o autor desenvolveu uma ontologia responsável por construir os planos de aula de forma automatizada.

Para integrar ao sistema SCIENCE [22] aulas de física com o uso de Robótica Educacional, foi desenvolvido uma ontologia por meio da ferramenta Protégé [20] denominada de Onto-Física. Essa ontologia consiste em descrever atividades envolvendo assuntos de física do ensino médio com a Robótica Educacional, a fim de promover aos professores instrução sobre o uso da robótica como ferramenta educacional.

De acordo com Almeida e Bax [1], as metodologias são desenvolvidas com o intuito de sistematizar a estruturação das ontologias, elas podem ser utilizadas para construção, aprendizado e integração do domínio e os conceitos que o relacionam. Desse modo as metodologias podem ser aplicadas de acordo com a funcionalidade e aplicação da ontologia. Dentre as metodologias desenvolvidas, pode-se citar a *Enterprise Model Approach*, *Methontology* e *Ontology Development 101*.

A fim de auxiliar na construção da ontologia foi utilizada a metodologia *Ontology Development 101* [21], que demonstra 7 passos necessários para o desenvolvimento da ontologia que podem ser observados na Figura 4.

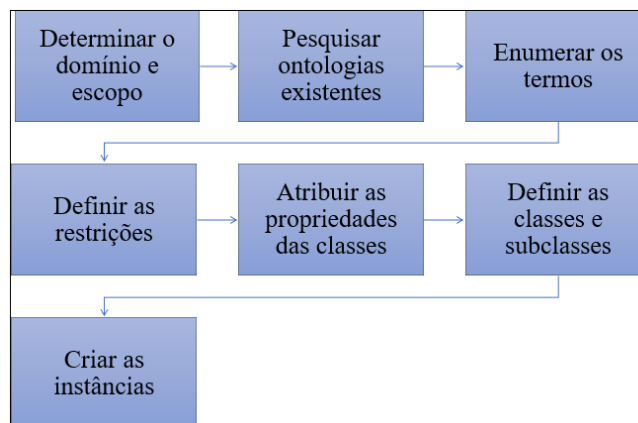


Figura 4. Metodologia *Ontology Development 101* (Autoria própria)

Para verificar se a ontologia está concisa e consistente foi utilizado o HerMiT [27] como mecanismo de inferência, que é uma ferramenta que fornece serviços de raciocínio de lógica de descrição para ontologias. Ele está disponível como uma biblioteca Java de código aberto, e pode processar ontologias de qualquer formato, como RDF/XML, OWL e OBO.

3.3.1. Definição do Domínio e Escopo da ontologia

De acordo com Duarte e Falbo [7], o domínio é o assunto de interesse da ontologia, na qual são descritos por conceitos e relações. Para a construção da Onto-Física o primeiro passo foi a definição do domínio, sendo ele os planos de aulas de física com uso de Robótica Educacional para alunos do Ensino Médio.

Com base na metodologia de Noy e McGuinness [21] para especificação de requisitos necessários para a construção de uma ontologia, foi realizado por meio da Tabela 1 um Documento de Especificação de requisitos para a Onto-Física, que expõe o objetivo, escopo, linguagem de implementação, usuários finais, caso de uso e requisitos da ontologia [22].

Tabela 1. Especificação de requisitos para a construção da ontologia. (Autoria Própria)

1	Objetivo
	Essa ontologia tem o objetivo de fornecer ao sistema SCIENCE [22], planos de aulas de física com o uso de Robótica Educacional, com atividades e programações para ser aplicados ao kit MINDSTORMS EV3 da Lego. Para assim promover aos professores um melhor entendimento acerca da ferramenta e difundir a utilização de Robótica Educacional nas escolas.
2	Escopo
	O foco principal e domínio dessa ontologia são os planos de aulas de física com a utilização de Robótica Educacional como recurso didático.
3	Linguagem de Implementação
	A linguagem que será utilizada é OWL (<i>Web Ontology Language</i>)
4	Usuários finais
	Os professores de física do ensino médio.
5	Caso de uso

Auxiliar na programação do robô.	
Auxiliar na construção de atividades interdisciplinares com robótica.	
Ajudar no uso de robótica como ferramenta didática nas aulas de Física.	
6	Requisitos da ontologia
I. Requisitos Não-Funcionais O vocabulário aplicado na ontologia deve ser retirado do contexto de Robótica Educacional.	
II. Requisitos Funcionais Q1 – Quais atividades podem ser utilizadas em aulas de física? Nessa questão, a ontologia deve retornar todas as atividades presentes que possam ser utilizadas em aulas de física.	
Q2 – Em uma determinada atividade, quais tarefas de programação podem ser utilizadas para a solução? Nessa questão, a ontologia deve indicar as tarefas de programação que promovem a resolução da atividade.	
Q3 – Quais atividades podem ser realizadas em determinada série do ensino médio? Para essa questão, a ontologia deve retornar as atividades disponíveis para uma determinada série, no caso dessa ontologia, irá retornar atividades do 1º e 2º ano do ensino médio.	

3.3.2. Enumeração de termos

Para organizar e pontuar os termos das descrições que a ontologia deve retornar a respeito do domínio é necessário satisfazer 3 perguntas [21], que são elas:

- Quais termos devem ser pontuados?
- Quais propriedades esses termos tem?
- Quais descrições cada termo deve fornecer?

Tendo em vista a necessidade de criar esses termos, para a ontologia Onto-Física, os seguintes termos foram identificados: Aula, Atividade, ComponenteCurricular, Subtarefas e Comandos.

3.3.3. Definição de classes e subclasses

As classes descrevem os conceitos do domínio. Dentro de uma classificação hierárquica as classes podem se subdividir em superclasses ou subclasses, sendo que as subclasses herdam as propriedades da superclasse [5]. Para a Onto-Física as classes que derivam da classe nativa owl:Thing são: *Atividade*, *Aula*, *Comandos*, *ComponenteCurricular* e *SubTarefas*.

A classe atividade representa as questões de física com a utilização de robótica educacional. A classe *Aula* representa os componentes que permitem a aula progredir que são o recurso didático necessário, o tempo de realização da aula, o componente curricular, o objetivo e a referência. A classe *Comandos* representa todos os blocos de programação do programa EV3 Classroom. A classe *ComponenteCurricular* representa os conteúdos de física que serão utilizados com a robótica. A classe *Subtarefas* representa tarefas de programação necessárias o aluno saber para solucionar a atividade.

A fim de expor a hierarquia de classes, na Figura 5, é demonstrado como as classes estão organizadas dentro da ontologia.

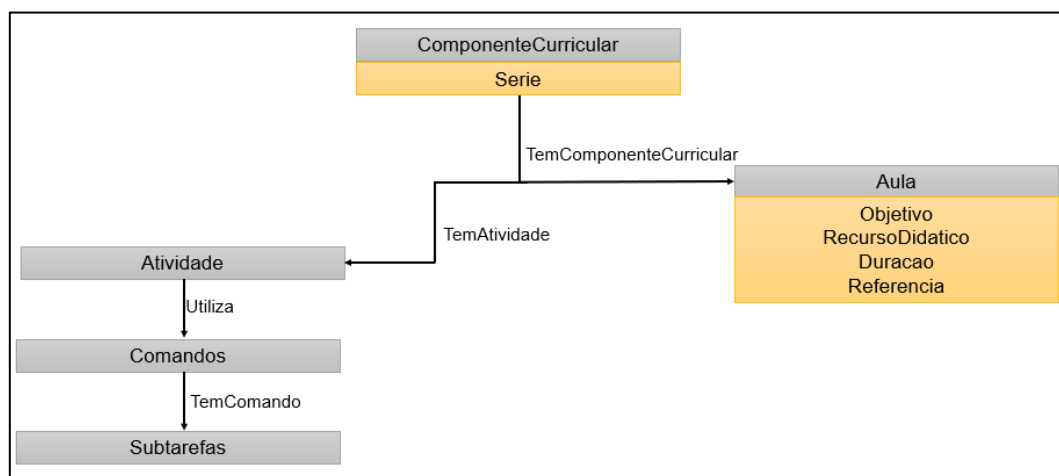


Figura 5. Hierarquia de classes (Autoria própria)

3.3.4. Definição das propriedades de classes

Para essa ontologia foram definidos dois tipos de propriedades, a *Object Property* e a *Datatype Property*. A primeira tem a função de qualificar ou relacionar classes e a segunda constitui campos que podem ser instanciados [22].

As propriedades do tipo *Object Property* criadas foram: *TemAtividade*, *TemComando*, *TemComponenteCurricular*, *TemSubtarefa*, *Utiliza*, *InAtividade*, *InComando*, *InComponenteCurricular* e *InSubtarefa*. Na Figura 6 é demonstrado a relação estabelecida entre as propriedades e as classes.

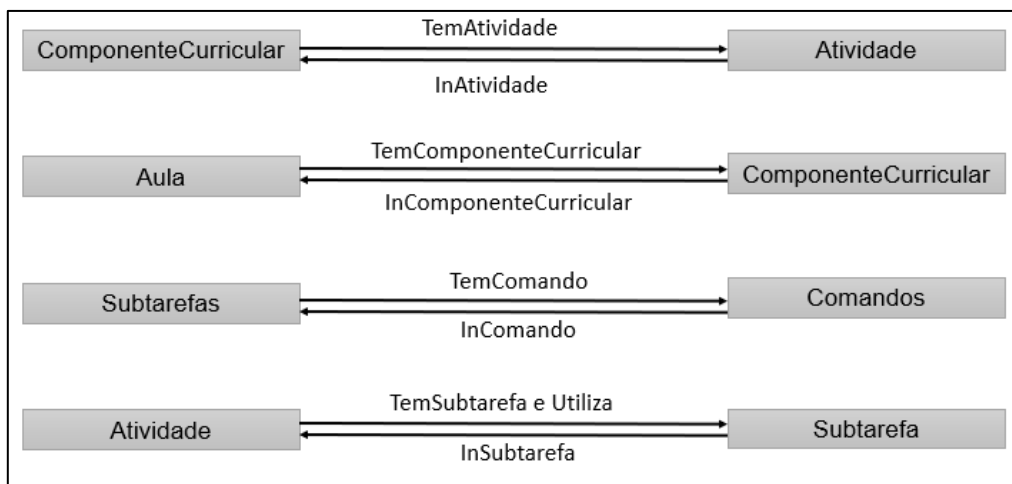


Figura 6. Relação entre classes e propriedades *Object Property* (Autoria própria)

Já as propriedades do tipo *Datatype Property* desenvolvidas, foram: *TemDuracao*, *TemObjetivo*, *TemQuantidade*, *TemRecursoDidatico*, *TemReferencia* e *TemSerie*. Na Figura 7 observa-se as relações que as classes tem com números inteiros e textos estabelecidas por essas propriedades.

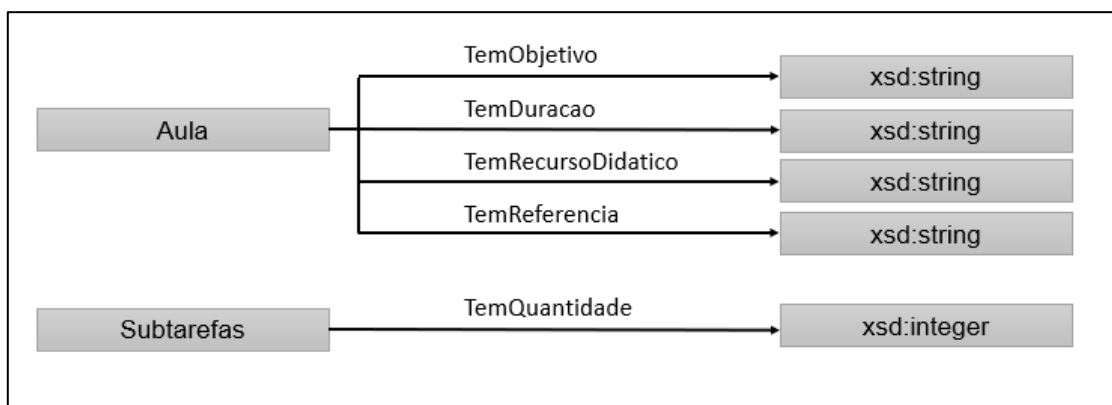


Figura 7. Relação entre classes e propriedades *Datatype Property* (Autoria própria)

3.3.5. Axiomas

Segundo Almeida e Bax [1], a base da ontologia são os conceitos, suas propriedades e relacionamentos, porém, para definir a semântica de seus termos, deve-se criar os axiomas, que são utilizados para modelar sentenças consideradas sempre verdadeiras. Passos [22] complementa apontando que para uma classe primitiva, utiliza-se axiomas precisos com *SubClassOf*, e para uma classe definida, utiliza-se axiomas precisos com *EquivalentTo*.

A seguir será demonstrado os axiomas desenvolvidos para cada classe, demonstrando as descrições dos conceitos estabelecidas para *Atividade*, *Aula*, *ComponenteCurricular* e *SubTarefas*.

Atividade

A classe *Atividade* é do tipo definida, ela é responsável por descrever as atividades que serão utilizadas pelo professor e as tarefas de programação necessária para do desenvolvimento das questões. Essa classe é composta pela classe *Subtarefas* que é responsável por interligar a classe *Atividade* com as tarefas de programação. A seguir no Quadro 1 são demonstradas as relações entre a classe *Atividade* e *Subtarefas* por meio dos axiomas.

Quadro 1. Descrição da Classe *Atividade* (Autoria própria)

1	Class: Atividade
2	SubClassOf:
3	utiliza some (TarefaExemplo1 and
4	(TemQuantidade value 1))
	utiliza some (TarefaExemplo2 and
	(TemQuantidade value 1))

Aula

A classe *Aula* é responsável por descrever o objetivo, duração, componente curricular, recurso didático e referência. Ela fornece as informações necessárias para que a aula possa acontecer. Por meio do Quadro 2 é possível analisar a relação existente entre a classe *Aula* e a classe *ComponenteCurricular*, como também a classe *Aula* com as propriedades de dados *TemComponenteCurricular*, *TemObjetivo*, *TemDuracao*, *TemRecursoDidatico* e *TemReferência*.

Quadro 2. Descrição da Classe *Atividade* (Autoria própria)

1	Class: Atividade
2	SubClassOf:
3	TemComponenteCurricular some ExemploAssunto
4	TemObjetivo value “ExemploObjetivo”
5	TemDuracao value “ExemploDuracao”
6	TemRecursoDidatico value “ExemploDuracao”
7	TemReferencia value “ExemploReferencia”

Subtarefas

A classe *Subtarefas* é responsável por fornecer a descrição de como utilizar os blocos de programação do EV3 Classroom de maneira correta para solucionar as questões da atividade. Essa classe se relaciona com a classe *Comandos* para relacionar a sub tarefa com um determinado bloco. Na Quadro 3 é demonstrado essa relação.

Quadro 3. Descrição da Classe *Subtarefas* (Autoria própria)

1	Class: Subtarefas
2	SubClassOf:
3	TemComando some ExemploComando1
4	TemComando some ExemploComando2

ComponenteCurricular

A classe *ComponenteCurricular* descreve os assuntos aplicados na ontologia sobre física para alunos do 1º e 2º ano do ensino médio. No quadro 4 mostra a relação existente entre a classe *ComponenteCurricular*, a classe *Atividade* e a propriedade de dados *TemSerie*.

Quadro 4. Descrição da Classe *ComponenteCurricular* (Autoria própria)

1	Class: ComponenteCurricular
2	SubClassOf:
3	TemAtividade some ExemploAtividade
4	TemSerie some “1º ano do ensino médio”

3.4. Desenvolvimento da API

Para a construção da API foi aplicada a linguagem Java no *software* Eclipse [16], com a utilização do *framework* *Spring Boot* que foi desenvolvido com o objetivo de simplificar a construção da programação na linguagem Java [24].

Para o desenvolvimento da API que poderá ser aplicada ao *back-end* do sistema SCIENCE, foi utilizado o sistema arquitetural RESTfull [17], separando a representação de dados em duas camadas, sendo elas, as camadas Modelo (*Model*) e Controle (*Controller*) [6]. Em conjunto foi utilizado o pacote *Data Access Object* (DAO) com o *framework* Apache Jena [24], que é responsável por executar e verificar a ontologia, salvar em formato

RDF/XML, e realizar as consultas SPARQL que retornam os dados necessários para a construção dos planos de aula. Para testar as requisições HTTP estabelecidas na camada de controle, foi utilizado o *software* Insomnia [14].

Na construção da API, a divisão dos pacotes Modelo e Controle ocorreram da seguinte forma:

- **Modelo (Model)**

A camada de Modelo é responsável por declarar os dados que serão utilizados para as aulas e programações, na qual esses dados são retirados da ontologia desenvolvida. Essa camada é composta por duas classes, a primeira é a denominada de “Aula.java”, que são definidos os atributos necessários para construção do plano de aula, sendo eles: Referência; Assunto; Conteúdo; Disciplina; Série; Duração; Recurso Didático; Objetivo; Atividade; Nome; Programação; e Descrição. A segunda classe é a “Programacao.java”, na qual são definidos os atributos relacionados a programação do robô, sendo eles: Bloco; Descrição do bloco; Atividade; Tarefa; e Descrição da tarefa.

- **Controle (Controller)**

A camada Controle apresenta uma classe chamada de “Ontofisica.java”, que é responsável por descrever as consultas por meio dos métodos GET, que faz a ligação entre as informações obtidas na ontologia por meio do DAO e os dados especificados na camada Modelo, direcionando as consultas corretamente para cada requisição.

Na Figura 8, é demonstrado a relação estabelecida entre as duas camadas.

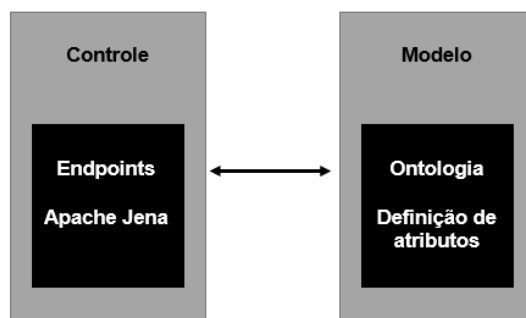


Figura 8. Diagrama de componentes da API. (Autoria Própria)

4. RESULTADOS

Para obter êxito no objetivo exposto no Capítulo 1, as atividades realizadas no trabalho ocorreram nas seguintes etapas: (1) Elaboração dos planos de aula; (2) Desenvolvimento da ontologia; (3) Construção da API; (4) Integração ao sistema SCIENCE.

No total foram desenvolvidos 6 planos de aulas para a disciplina de Física com o uso de Robótica Educacional, seguindo o mesmo formato e layout dos planos de aula desenvolvidos por Passos [22], para padronizar todos os planos de aula presentes no sistema. Os conteúdos abordados nos planos de aula foram: Velocidade Média; Aceleração; Conversão de rotações para centímetros; Intensidade da luz refletida; Ondas unidimensionais; e Lançamento Horizontal.

Com a definição das classes, propriedades e axiomas, foram estabelecidas todas as relações e especificações visando estruturar o domínio de interesse da ontologia, na Figura 9 é possível visualizar a estrutura da ontologia e como as classes se relacionam com o domínio.

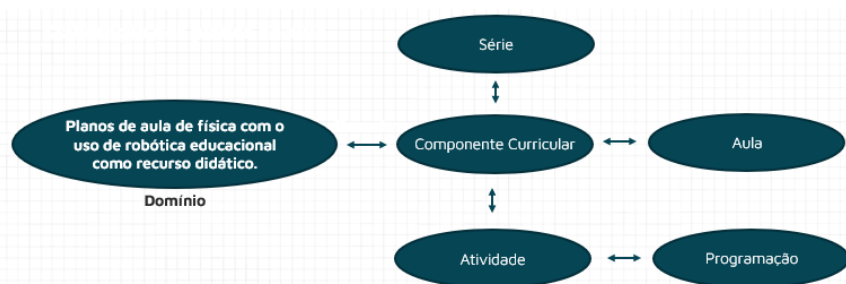


Figura 9. Estrutura com relações entre as classes e o domínio. (Autoria Própria)

4.1 Consultas SPARQL na ontologia

Na ontologia desenvolvida por meio do *software* Protégé, foram definidas todas as classes, instâncias, propriedades e axiomas relacionados com os planos de aula. Para consultar o resultado final da ontologia, foi realizada a verificação em duas etapas, a primeira por meio do mecanismo de inferência HermiT, que demonstrou que a ontologia está concisa e estruturada corretamente.

A segunda etapa ocorreu por meio das consultas realizadas utilizando a linguagem SPARQL. Segundo Penteadó [23], o SPARQL (*SPARQL Protocol and RDF Query Language*) para consultas em bases RDF é a linguagem padrão, na qual os planos de execução de consulta são gerados a partir do conhecimento prévio de como uma base de dados RDF foi particionada. Passos [22] complementa mencionando que as questões principais que a ontologia deve responder devem ser descritas e formalizadas por meio da linguagem de consulta SPARQL.

Dito isto, as consultas principais que foram realizadas na ontologia Onto-Física são: (1) “Quais planos de aula podem ser utilizados nos componentes curriculares de Física no Ensino Médio?”; (2) “Quais blocos de programação são utilizados para solucionar determinada atividade?”; (3) “Quais subtarefas podem ser utilizadas para aprender as funcionalidades dos blocos de programação?”.

Para a primeira pergunta, a ontologia retornou por meio da consulta todos os itens necessários que compõe a aula de robótica, que são: a disciplina, o assunto, a descrição da atividade, a descrição da sub tarefa de programação, a série, a referência, a duração e o objetivo. No quadro 5, descreve-se o código da consulta utilizada para retornar a pergunta (1).

Quadro 5. Quais planos de aula podem ser utilizados nos componentes curriculares de Física no Ensino Médio?
(Autoria Própria)

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ont: <http://www.semanticweb.org/usuario/ontologies/2021/8/untitled-ontology-7#>
SELECT DISTINCT ?Disciplina ?Assuntos ?Descricao ?conteudo ?AtividadeEspecificas ?Serie
?Programacao ?RecursoDidatico ?Referencia ?Duracao ?Objetivo ?Aula
WHERE {
    #Aula
    ?Aula rdfs:subClassOf ?subclasseaula.
    ?subclasseaula owl:someValuesFrom ?conteudo.

    #Conteudo
    ?conteudo rdfs:subClassOf* ont:ComponenteCurricular.

    #Assunto
    ?conteudo rdfs:subClassOf ?Assuntos.
    ?Assuntos rdfs:subClassOf* ?Disciplina.

    #Disciplina
    ?Disciplina rdfs:subClassOf ont:ComponenteCurricular.

    #AtividadeEspecificas
    ?conteudo rdfs:subClassOf ?atividade.
    ?atividade owl:someValuesFrom ?AtividadeEspecificas.
    ?AtividadeEspecificas rdfs:comment ?Descricao.

    #Recurso Didatico
    ?Aula rdfs:subClassOf ?x.
    ?x owl:hasValue?RecursoDidatico.
    ?x owl:onProperty ont:TemRecursoDidatico.

    #Referencia
    ?Aula rdfs:subClassOf ?y.
    ?y owl:hasValue?Referencia.
    ?y owl:onProperty ont:TemReferencia.

    #Duracao
    ?Aula rdfs:subClassOf ?z.
    ?z owl:hasValue?Duracao.
    ?z owl:onProperty ont:TemDuracao.

    #Objetivo?Aula rdfs:subClassOf ?w.
    ?w owl:hasValue?Objetivo.
    ?w owl:onProperty ont:TemObjetivo.
```

```

#Serie
?conteudo rdfs:subClassOf ?a.
?a owl:hasValue ?Serie.
?a owl:onProperty ont:TemSerie.

#Programacao
?AtividadeEspecificas rdfs:subClassOf ont:Atividade.
?AtividadeEspecificas rdfs:subClassOf ?t.
?t owl:someValuesFrom ?b.
?b owl:intersectionOf ?c.
?b owl:intersectionOf ?e.
?c rdf:rest*/rdf:first ?Programacao.
?e rdf:rest*/rdf:first ?f.
?f owl:hasValue ?quan.
FILTER NOT EXISTS {?Programacao owl:onProperty ont:TemQuantidade}
}

```

Na segunda pergunta, a ontologia retornou os blocos de programação utilizados em cada atividade envolvendo Robótica Educacional, levando em consideração os assuntos de física descritos na ontologia, deve ser apresentado também a descrição dos blocos de programação. No Quadro 6, descreve-se o código desenvolvido para essa consulta.

Quadro 6. Quais blocos de programação são utilizados para solucionar determinada atividade? (Autoria própria)

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ont: <http://www.semanticweb.org/usuario/ontologies/2021/8/untitled-ontology-7#>
SELECT ?atividade ?comandos ?descricao
WHERE {
  {
    ?atividade rdfs:subClassOf ont:Atividade.
    ?atividade rdfs:subClassOf ?x.
    ?x owl:someValuesFrom ?a.
    ?a owl:intersectionOf ?b.
    ?b rdf:first ?c.
    ?c rdfs:subClassOf ?d.
    ?d owl:someValuesFrom ?comandos.
    ?comandos rdfs:comment ?descricao.
    ?a owl:intersectionOf ?f.
    ?f rdf:rest*/rdf:first ?g.
    ?g owl:hasValue ?t
  }
  UNION
  {
    ?atividade rdfs:subClassOf ont:Atividade.
    ?atividade rdfs:subClassOf ?x.
    ?x owl:someValuesFrom ?a.
    ?a rdfs:subClassOf ?b.
    ?b owl:someValuesFrom ?comandos.
    ?comandos rdfs:comment ?descricao
  }
}

```

Já para a terceira pergunta, a consulta retornou os blocos de programação utilizados para solucionar a atividade e subtarefas que demonstram como utilizar esses blocos. No Quadro 7 é descrito o código utilizado para realização da consulta.

Quadro 7. Quais subtarefas podem ser utilizadas para aprender as funcionalidades dos blocos de programação? (Autoria própria)

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX ont: <http://www.semanticweb.org/usuário/ontologies/2021/8/untitled-ontology-7#>
SELECT DISTINCT ?atividade ?tarefa ?descricaodatarefa ?comandos ?descricao
WHERE {
  {
    ?atividade rdfs:subClassOf ont:Atividade.
    ?atividade rdfs:subClassOf ?x.
    ?x owl:someValuesFrom ?a.
    ?a owl:intersectionOf ?b.
    ?b rdf:first ?tarefa.
    ?tarefa rdfs:subClassOf ?d.
    ?tarefa rdfs:comment ?descricaodatarefa.
    ?d owl:someValuesFrom ?comandos.
    ?comandos rdfs:comment ?descricao.
    ?a owl:intersectionOf ?f.
    ?f rdf:rest*/rdf:first ?g.
  }
  UNION
  {
    ?atividade rdfs:subClassOf ont:Atividade.
    ?atividade rdfs:subClassOf ?x.
    ?x owl:someValuesFrom ?a.
    ?a rdfs:subClassOf ?b.
    ?b owl:someValuesFrom ?comandos.
    ?comandos rdfs:comment ?descricao.
  }
}

```

Com base nas informações retornadas nas consultas SPARQL, foi possível verificar que a ontologia consegue responder todas as 3 perguntas definidas de maneira automatizada.

4.2 Visualização das informações obtidas por meio da API

A API⁷ desenvolvida para geração de planos de aula de física com o uso de Robótica Educacional, que poderá ser integrada ao *back-end* do sistema SCIENCE, foi construída seguindo o mesmo formato e *layout* dos planos de aula desenvolvidos por Passos [22]. Com relação a parte de visualização do usuário final, nesse trabalho não foram realizadas ações no *front-end* do sistema.

A fim de visualizar o resultado final das informações resgatadas pela API e testar as requisições HTTP estabelecidas na camada de controle por meio dos *endpoints* expostos na Tabela 2, foi utilizado o *software* Insomnia [14] para visualização dos resultados. Com o uso dessa ferramenta, foi possível verificar as consultas de modo a retornar todos os atributos necessários para a formação dos planos de aula, para isso foi utilizado as consultas SPARQL expostas nos Quadros 5, 6 e 7. Com base nos resultados obtidos na consulta, foi concluído que a API fornece todas as informações de forma correta para os planos de aula, retornando a série, duração da aula, conteúdo, assunto, programação, recursos didáticos e atividade.

Tabela 2. *Endpoints* desenvolvidos para realizar consultas na ontologia. (Autoria Própria)

Endpoints	Descrição
@GetMapping("/query")	Consultar todas as informações necessárias para construção do plano de aula, que são: o conteúdo, o assunto, a disciplina, a série, o recurso didático, o objetivo, a atividade de programação e a referência.
@GetMapping("/programacaolist")	Consulta que resgata todas as tarefas de programação presentes na ontologia.
@GetMapping("/programacao")	Retorna todas as tarefas de programação necessárias para solucionar as questões da atividade. Nessa requisição é necessário especificar a atividade de programação que deseja consultar.

⁷ Disponível em: <https://bityli.com/tCHmYR>

@GetMapping("/assuntos")	Consultar os assuntos disponíveis de física para determinada série. Por exemplo, para o 2º ano do Ensino Médio os assuntos disponíveis são Ondas e Reflexão de Luz.
@GetMapping("/serie")	Consultar quais séries apresentam planos de aula na ontologia. Na consulta desse <i>endpoint</i> deve retornar o 1º e 2º ano do Ensino Médio.
@GetMapping("/comandos")	Realizar uma listagem de todos os blocos de programação do <i>software</i> EV3 Classroom e a descrição de cada um deles.
@GetMapping("/conteudosAssunto")	Consulta que retorna os conteúdos, assuntos e série presentes na ontologia.
@GetMapping("/conteudos")	Esse <i>endpoint</i> é responsável por consultar e listar os conteúdos disponíveis para determinada série e assunto.
@GetMapping("/buscaAvancada")	Realiza uma busca avançada na ontologia, na qual deve-se especificar a série, o conteúdo e o assunto, retornando o plano de aula que apresenta os três parâmetros.
@GetMapping("/busca")	Realiza uma busca geral na ontologia, por exemplo, os planos de aula com o assunto de Cinemática.

Na Figura 10, utilizando o *endpoint* "@GetMapping("/programacao")", exemplifica-se o resultado de uma consulta realizada no *software* Insomnia para listar as tarefas de programação necessárias para solucionar as questões presentes na atividade do conteúdo de Lançamento Horizontal, na qual foi utilizada a consulta SPARQL presente no Quadro 7 para resgatar na ontologia os dados necessários. Os atributos utilizados para essa consulta foram: *bloco*, que expõe o bloco de programação que será utilizado; *descript*, na qual descreve a funcionalidade daquele bloco; *Atividade*, que especifica em qual atividade aquele bloco será utilizado; *tarefa*, onde é especificado a tarefa de programação que será utilizada; e *descTarefa*, que descreve a tarefa de programação, auxiliando a entender como funciona o bloco.

```

1 = [
2   {
3     "bloco": "MoverDirecaoDuracaoVelocidade",
4     "descript": "Bloco Mover com Direção por uma Duração em Velocidade\n\nDescrição:
Move o robô por uma quantidade específica de rotações, graus ou segundos, em uma
determinada direção e velocidade.\n\nÉ correto utilizar o bloco de definir motores
de movimento antes desse bloco.\n\nLink da imagem do bloco:
https://imgur.com/ueUYeXs",
5     "atividade": "AtividadeLancamentoHorizontal",
6     "tarefa": "TarefaMoverDuracaoVelocidade",
7     "descTarefa": "TAREFA DE MOVER O ROBÔ DEFININDO VELOCIDADE E DURAÇÃO\n\n-
Objetivo do Objetivo da Lição\n\n1. Aprender a fazer o robô locomover-se definindo a
velocidade e a duração do deslocamento.\n\nPROCEDIMENTO\n\n1. Etapa: Abre a aba de
Movimento.\n2. Etapa: Coloque o bloco \"Iniciar Movimento com Duração a uma
Velocidade de\" na sua pilha de programação.\n3. Etapa: Defina a direção do
deslocamento, em 0, o robô irá reto.\n4. Etapa: Defina a duração em rotação, graus
ou segundos.\n5. Etapa: Defina a velocidade de deslocamento em percentual."
8   },
9   {
10    "bloco": "Inicio",
11    "descript": "Bloco Quando Programa Iniciar\n\nDescrição: Deve ser sempre o
primeiro bloco utilizado em uma programação, pois permite executar os blocos que vem
na sequência.\n\nLink da imagem do bloco: https://imgur.com/g4vx06X",
12    "atividade": "AtividadeLancamentoHorizontal",
13    "tarefa": "TarefaIniciar",
14    "descTarefa": "TAREFA DE INICIAR\n\n- Objetivo do Objetivo da Lição\n\n1.
Aprender a fazer o robô iniciar uma programação.\n\nPROCEDIMENTO\n\n1. Etapa: Abre
a aba de Eventos.\n2. Etapa: Coloque o bloco \"Quando Programa Iniciar\" no inicio
da sua programação."
15  }
16 ]

```

Figura 10. Requisição para listar as tarefas de programação utilizada na atividade de Lançamento Horizontal. (Insomnia)

5. CONCLUSÃO

As atividades desenvolvidas nesse trabalho ocorreram visando auxiliar os professores de Física a utilizar a robótica educacional como ferramenta de ensino, pois foi identificado uma dificuldade por parte dos professores em utilizar a ferramenta. Diante disso, a ontologia e a API construídas para serem implementadas ao sistema SCIENCE são responsáveis fornecer aos docentes informações e instruções sobre como utilizar a Robótica Educacional durante as aulas, por meio de atividades e explicações de funcionamento dos blocos de programação.

O desenvolvimento da ontologia ocorreu a partir da utilização da metodologia *Ontology Development 101*, na qual foram definidas todas as classes, propriedades, axiomas necessários para formação dos planos de aula. Já a verificação da ontologia ocorreu por meio do mecanismo de inferência *HermiT* e por consultas realizadas em linguagem SPARQL.

A construção do API foi realizada por meio da utilização do pacote DAO e das camadas de Modelo e Controle. A partir da verificação de requisições por meio do *software* *Insomnia*, foi possível verificar e validar o resultado final das informações fornecidas pela API, na qual ela gera planos de aulas de física com os seguintes componentes: atividades, tarefas de programação, série, assunto, componente curricular, blocos de programação, descrição dos blocos de programação e objetivo.

As atividades desenvolvidas no trabalho ocorreram seguindo uma ordem cronológica de seis passos que são expostos na Figura 11.

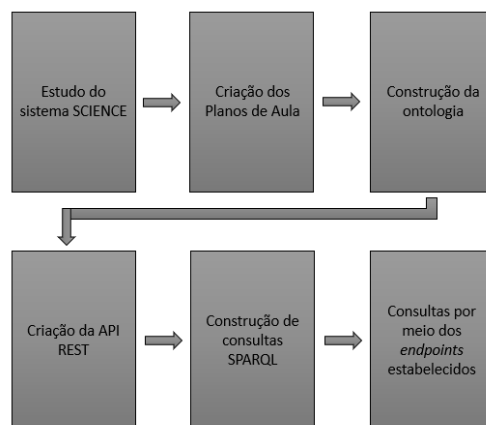


Figura 11. Esquema com as atividades desenvolvidas no trabalho. (Autoria Própria)

A principal contribuição desse trabalho é fornecer aos docentes instrução sobre o uso de robótica educacional em aulas de física, porém há um fator limitante relacionado a visualização dos planos de aula no sistema, pois foi desenvolvido a criação de planos de aulas e a API para possível implementação ao *back-end* do sistema SCIENCE.

No presente momento o sistema SCIENCE dispõe de 36 aulas de matemática do Ensino Fundamental II, e nesse trabalho houve a expansão para mais 6 aulas de física do Ensino Médio. Com o propósito de ampliar o sistema fornecendo mais materiais aos professores, os seguintes trabalhos futuros são propostos:

- Criação de mais planos de aulas de Física com outros assuntos para somar ao sistema SCIENCE;
- Desenvolvimento de outras ontologias visando fornecimento de planos de aulas para outras disciplinas, como por exemplo, Biologia, Português, História e Geografia.
- Desenvolvimento da etapa de visualização dos planos de aula de Física no *front-end* do sistema SCIENCE, na qual será utilizada a API criada nesse trabalho para obter as informações necessárias sobre os planos de aulas.

6. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ALMEIDA, Mauricio B.; BAX, Marcello P. Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da informação*, v. 32, p. 7-20, 2003.
- [2] BARROSO, Marta F.; RUBINI, Gustavo; SILVA, Tatiana da. Dificuldades na aprendizagem de Física sob a ótica dos resultados do Enem. *Revista Brasileira de Ensino de Física*, São Paulo, v. 40, n. 4, 18 jun. 2018.
- [3] BRASIL. Base Nacional Comum Curricular. Ministério da Educação, 2021. Disponível em: < http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf >. Acesso em: 08 nov. 2021
- [4] CAMPOS, Flavio Rodrigues. Robótica Educacional no Brasil: questões em aberto, desafios e perspectivas futuras. *Revista ibero-americana de estudos em educação*, v. 12, n. 4, p. 2108-2121, 2017.
- [5] CARLAN, Eliana. Ontologia e Websemântica. 2006. Trabalho de conclusão de curso (Bacharelado em Biblioteconomia) - Universidade de Brasília, Brasília.
- [6] COUTO, Rodolpho Sbaraglini; FOSCHINI, Ivan João. Análise comparativa entre dois Frameworks MVC para a Plataforma Java EE: JSF e VRaptor. *Revista TIS*, v. 4, n. 3, 2016.
- [7] DUARTE, Katia Cristina; FALBO, Ricardo de Almeida. Uma ontologia de qualidade de software. In: *Workshop de Qualidade de Software*, João Pessoa. 2000. p. 275-285.
- [8] FERNANDEZ, M., GOMEZ-PEREZ, A. and JURISTO, N. (1997) "METHONTOLOGY: From Ontological Art Towards Ontological Engineering", *AAAI-97 Spring Symposium on Ontological Engineering*, Stanford University, March 24-26th.

-
- [9] GARCIA, Leonardo da Silva. Experimentos no ensino de física utilizando a robótica lego® EV3 no ensino médio e fundamental. 2018. Dissertação (Mestrado em Física) – Universidade Federal Rural do Semi-Árido, Mossoró.
- [10] GLORINHA, Martini *et al.* Conexões com a Física. 3. ed. São Paulo: Moderna, 2016.
- [11] GOMEZ-PEREZ, A., FERNANDEZ, M. and DE VICENTE, A.J. (1996) “Towards a Method to Conceptualize Domain Ontologies”, ECAI-96 Workshop on Ontological Engineering, Budapest.
- [12] GRUBER, Thomas R. Toward principles for the design of ontologies used for knowledge sharing?. International journal of human-computer studies, v. 43, n. 5-6, p. 907-928, 1995.
- [13] GUARINO, N. Formal ontology in information systems. In: IOS Press, A, editor, FOIS’98, TRENTO, ITALY, p. 3–15, 06 1998.
- [14] INSOMNIA. 2021. Disponível em: <<https://insomnia.rest/>>. Acesso em: 30 out. 2021.
- [15] JONES, Dean; BENCH-CAPON, Trevor; VISSER, Pepijn. Methodologies for ontology development. 1998.
- [16] LUCKOW, Décio Heinzelmann; DE MELO, Alexandre Altair. Programação Java para a WEB. Novatec Editora, 2010.
- [17] MALIPENSE, Leticia Miranda; ZUCHI, Jederson Donizete. UM ESTUDO SOBRE O CONCEITO E A APLICAÇÃO DA ARQUITETURA DE MICROSERVIÇOS. Revista Interface Tecnológica, v. 15, n. 1, p. 122-134, 2018.
- [18] MINDSTORMS. 2021. Disponível em: <<https://www.lego.com/pt-pt/themes/mindstorms/ev3>>. Acesso em: 30 out. 2021.
- [19] MORAIS, Edison Andrade Martins; AMBRÓSIO, Ana Paula L. Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens. Relatório Técnico–RT-INF-001/07, 2007.
- [20] MUSEN, M. A. The protégé project: a look back and a look forward. AI matters, ACM, v. 1, n. 4, p. 4–12, 2015.
- [21] NOY, Natalya F; MCGUINNESS, Debora L. Ontology development 101: A guide to creating your first ontology. 2001.
- [22] PASSOS, Daniel Deyson Nunes. Incentivo a popularização da robótica educacional como recurso de ensino por meio da utilização de um sistema de geração automatizada de planos de ensino. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal Rural do Semi-Árido, Mossoró, 2021.
- [23] PENTEADO, Raqueline Ritter de Moura. Otimização de consultas SPARQL em bases RDF distribuídas. 2017. Tese (Doutorado em Ciência da Computação) - Universidade Federal do Paraná, Curitiba, 2017.
- [24] LECHETA, Ricardo R. Web Services RESTful: Aprenda a criar web services RESTful em Java na nuvem do Google. Novatec Editora, 2015.
- [25] SANTOS, Carmen Faria; MENEZES, Crediné Silva de. A Aprendizagem da Física no Ensino Fundamental em um Ambiente de Robótica Educacional. In: Anais do Workshop de Informática na Escola, 2005.
- [26] SEGUNDO, José Eduardo Santarém. Web Semântica: Introdução a recuperação de dados usando SPARQL. Encontros Nacionais de Pesquisa e Pós-Graduação em Ciência da Informação. 2017.
- [27] SHEARER, Rob; MOTIK, Boris; HORROCKS, Ian. Hermit: A Highly-Efficient OWL Reasoner. In: Owled. 2008. p. 91.
- [28] SILVA, Alzira Ferreira da. RoboEduc: Uma Metodologia de Aprendizado com Robótica Educacional. 2009. Tese (Doutorado em Ciências) - Universidade Federal do Rio Grande do Norte, Natal.
- [29] ZILLI, Silvana do Rocho. A robótica educacional no ensino fundamental: perspectivas e prática. Orientador: Ana Maria Franzoni. 2004. Dissertação (Mestrado em Engenharia de Produção) - Universidade Federal de Santa Catarina, Santa Catarina, 2004.



MINISTÉRIO DA EDUCAÇÃO
Universidade Federal Rural do Semi-Árido – UFERSA
Centro de Ciências Exatas e Naturais – CCEN

ATA DE DEFESA DE TRABALHO DE CONCLUSÃO DE CURSO

Às dez horas e trinta minutos do dia dezessete de novembro de dois mil e vinte e um, na plataforma virtual Google Meet, reuniu-se a Banca Examinadora de defesa de trabalho de conclusão de curso de autoria do aluno **JOÃO TADAYOSHI TATSUKAWA NETO**, aluno do curso de Bacharelado em Ciência e Tecnologia desta universidade, N° de matrícula **2018010766**, com o título **“DESENVOLVIMENTO DE UMA ONTOLOGIA SOBRE PLANOS DE AULA DE FÍSICA COM O USO DA ROBOTICA EDUCACIONAL PARA APLICAR AO SISTEMA SCIENCE”**. A Banca Examinadora ficou assim constituída por quatro membros: Prof. Dr. SILVIO ROBERTO FERNANDES DE ARAÚJO, presidente da banca e orientador do Trabalho de Conclusão de Curso; Prof. Dr. PAULO GABRIEL GADELHA DE QUEIROZ, coorientador, Prof. Dra. AMANDA GONDIM DE OLIVEIRA e Bsc. RODOLFO FELIPE MEDEIROS ALVES, como membros. Concluída a defesa, procedeu-se o julgamento pelos membros da banca examinadora, em reunião fechada, tendo o aluno sido considerado APROVADO. E para constar, eu, SILVIO ROBERTO FERNANDES DE ARAÚJO, lavrei a presente ata que, após lida e aprovada pelos membros da banca examinadora, será assinada por todos.

Mossoró, 17 de Novembro de 2021.

Assinatura dos membros da Banca Examinadora.

Prof. Dr. SILVIO ROBERTO FERNANDES DE ARAÚJO – UFERSA

Presidente e orientador

Prof. Dr. PAULO GABRIEL GADELHA DE QUEIROZ – UFERSA

Primeiro Membro e coorientador

AMANDA GONDIM DE
OLIVEIRA:01267843454

Profa. Dra. AMANDA GONDIM DE OLIVEIRA - UFERSA

Segundo Membro

Documento assinado digitalmente

gov.br
Bsc.

RODOLFO FELIPE MEDEIROS ALVES
Data: 17/11/2021 15:17:37-0300
Verifique em <https://verificador.itl.br>

LVES

TERCEIRO MEMBRO