

# Sistemas Especialistas

**Autores: Matheus Mello e Jorge Harbes<sup>1</sup>**

<sup>1</sup>Centro Federal de Educação Tecnológica Celso Suckow da Fonseca - CEFET/RJ

jharbes@hotmail.com

matheusmello142012@gmail.com

**Resumo.** *Este artigo tem o propósito de reunir informações sobre as experiências e conhecimentos absorvidos nas aulas de sistemas especialistas, lecionada pelo professor Carlos Eduardo Pantoja.*

## 1. Introdução

Nesta disciplina aprendemos um pouco mais sobre agentes, sistemas multiagentes, agentes cognitivos baseados em BDI e programação baseada em estímulos. O conteúdo é vasto, e as possibilidades, exponenciais. Como, por exemplo, um novo modelo de comunicação entre drones, abrindo novas possibilidades para esta tecnologia que hoje já se tornou um mercado.

Utilizamos a IDE ChonIDE, desenvolvida pelos nossos próprios colaboradores.

## 2. O que é um agente ?

Um agente é uma entidade autônoma e computacional que pode perceber seu ambiente, tomar decisões, agir e interagir com outros agentes para alcançar um ou mais objetivos. Normalmente é programado para que, diante de determinada análise, tome decisões baseado em conhecimentos preestabelecidos ou adquiridos. É o protagonista de todo o enredo de sistema multiagente.

## 3. Sistema MultiAgente

Um sistema multiagente é composto por vários agentes que interagem entre si para atingir objetivos individuais ou coletivos. Estes agentes podem ser tanto cooperativos quanto competitivos, dependendo da natureza do sistema e dos objetivos a serem alcançados. Cada agente em um sistema multiagente possui a capacidade de operar de forma independente, mas a interação entre os agentes é crucial para o funcionamento eficaz do sistema como um todo.

## 4. Agentes cognitivos em BDI

BDI é uma abreviação de "Belief-Desire-Intention". É um modelo de arquitetura de agentes que é baseado nos conceitos de crenças (beliefs), desejos (desires) e intenções (intentions) de um agente.

- Crenças (Beliefs): São as informações que o agente possui sobre o mundo ao seu redor.
- Desejos (Desires): São os objetivos que o agente deseja alcançar.
- Intenções (Intentions): São os planos de ação que o agente decide executar para alcançar seus desejos.

## 5. O que é AgentSpeak?

AgentSpeak é uma linguagem de programação lógica voltada para a modelagem e implementação de sistemas multiagentes. Ela se baseia em lógica de ação, permitindo que agentes autônomos expressem suas crenças, intenções e ações de maneira formal, facilitando a comunicação e coordenação entre agentes em ambientes complexos. É amplamente utilizada em pesquisa em inteligência artificial distribuída e sistemas multiagentes. Foi criada na década de 1990 por Michael Wooldridge, Nicholas R. Jennings e David Kinny. Ela foi desenvolvida como uma linguagem de programação específica para sistemas multiagentes e desde então tem sido usada em pesquisas e aplicações na área de inteligência artificial distribuída e sistemas multiagentes.

## 6. O que é Jason?

Jason é uma implementação do interpretador AgentSpeak, uma linguagem orientada a agentes baseada em lógica, facilitando a criação de agentes inteligentes dentro de sistemas multiagentes (MAS). Influenciada pelo Prolog, essa ferramenta é particularmente adequada para desenvolver agentes seguindo o modelo BDI (Belief-Desire-Intention), onde os agentes possuem crenças sobre o mundo, desejos ou objetivos a serem alcançados, e intenções ou planos para alcançar esses desejos. Jason proporciona um ambiente de desenvolvimento que assiste na codificação, depuração e teste de agentes, e quando integrado aos frameworks CArtAgO e MOISE através da plataforma JaCaMo, oferece uma solução robusta para desenvolver e organizar sistemas multiagentes complexos.

# Experiência Realizada

## 1. Objetivo

O objetivo deste projeto é desenvolver um chatbot de nível básico, que incorpora os conceitos e técnicas aprendidos na disciplina de Sistemas Especialistas. O foco será em aplicar os fundamentos de linguagens de programação e agentes inteligentes, conforme discutido nas seções anteriores do documento. Especificamente, vamos explorar a implementação de Agentes Cognitivos baseados no modelo BDI (Belief-Desire-Intention), mas em um contexto mais simplificado e direto. O uso de sistemas multiagentes ajudará a estruturar o chatbot de forma que ele possa operar eficientemente, embora em um ambiente menos complexo do que os sistemas avançados.

## 2. Metodologia

A metodologia para o desenvolvimento do chatbot é fundamentada nos princípios e técnicas discutidos nas seções anteriores, envolvendo uma abordagem sistemática e faseada. O processo será dividido em várias etapas principais:

1. **Planejamento e Design:** Esta fase envolve a definição clara dos objetivos do chatbot, o escopo de suas funcionalidades e a estruturação básica de como os agentes interagirão no sistema. Será desenvolvido um design conceitual do chatbot, incluindo um diagrama de como os agentes BDI irão operar e interagir dentro do sistema.

2. **Desenvolvimento dos Agentes:** Utilizando a linguagem AgentSpeak, programaremos os agentes com capacidades específicas de percepção, raciocínio e ação. Cada agente será desenhado para realizar funções específicas dentro do chatbot, seguindo o modelo BDI, onde suas crenças, desejos e intenções guiarão seu comportamento.
3. **Integração e Testes:** Os agentes desenvolvidos serão integrados em um ambiente de sistema multiagente, usando a plataforma Jason. Esta etapa envolve a coordenação e comunicação entre os agentes, assegurando que eles funcionem harmoniosamente. Testes iterativos serão realizados para identificar e corrigir problemas de interação e funcionalidade.
4. **Interface de Usuário:** Desenvolvimento de uma interface de usuário amigável para permitir a interação efetiva entre o usuário e o chatbot. Esta interface será projetada para ser intuitiva, garantindo que os usuários possam se engajar facilmente com o chatbot.
5. **Avaliação e Iteração:** Após a implementação, o chatbot será submetido a uma série de avaliações para testar sua eficácia, usabilidade e precisão nas respostas. Com base no feedback recebido, faremos as iterações necessárias para melhorar e refinar o chatbot.

Essa abordagem metodológica garantirá que todos os aspectos do chatbot sejam cuidadosamente considerados e desenvolvidos, resultando em uma ferramenta interativa e eficaz que alinha os conceitos teóricos com a aplicação prática.

### 3. Implementação

A implementação do chatbot baseia-se nos princípios de sistemas multiagentes e na arquitetura BDI (Belief-Desire-Intention), utilizando a linguagem AgentSpeak e a plataforma Jason. O processo de implementação segue as seguintes etapas:

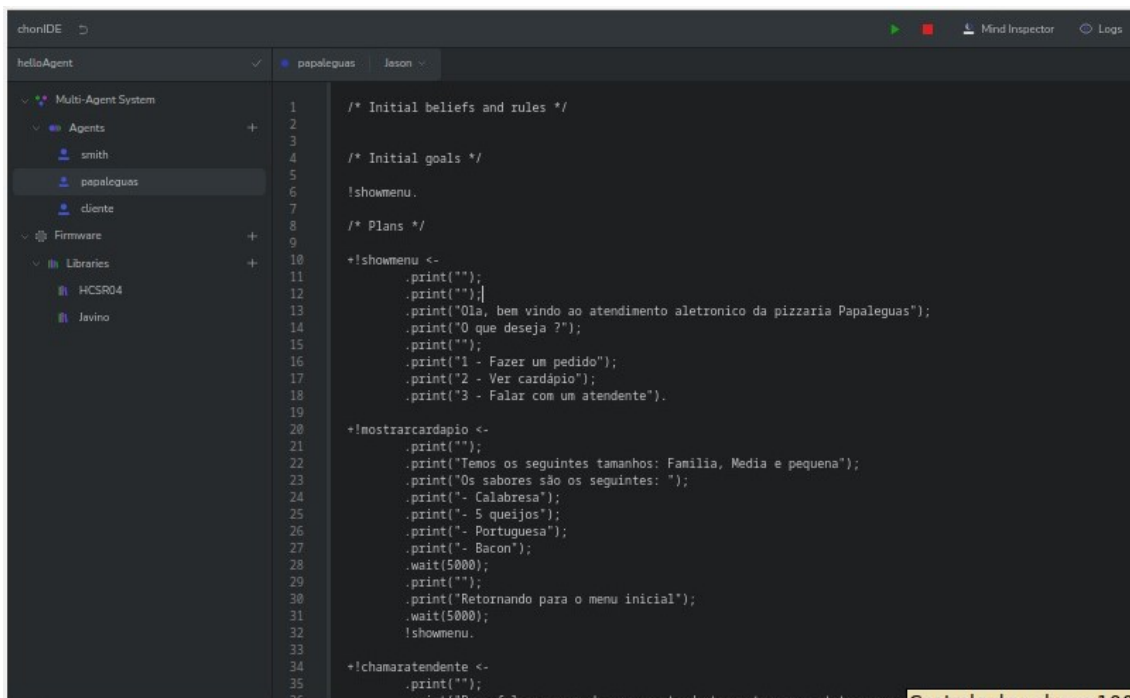
1. **Configuração do Ambiente de Desenvolvimento:** Inicialmente, configuramos o ambiente de desenvolvimento integrando a IDE (como ChonIDE ou similar) com a plataforma Jason. Isso nos permite desenvolver e testar os agentes em um ambiente controlado.
2. **Criação dos Agentes BDI:** Cada agente é criado com um conjunto específico de crenças (beliefs), desejos (desires) e intenções (intentions). As crenças representam o conhecimento do agente sobre o mundo e o estado do chatbot, os desejos indicam os objetivos que o agente busca alcançar, e as intenções são os planos que o agente executa para satisfazer seus desejos.
3. **Programação em AgentSpeak:** Utilizando AgentSpeak, programamos os comportamentos e as interações dos agentes. Isso inclui a capacidade de processar entradas do usuário, realizar raciocínio lógico, e responder apropriadamente. Os agentes são programados para serem autônomos, mas também capazes de comunicar-se e colaborar quando necessário.
4. **Integração dos Agentes:** Os agentes são integrados em um sistema multiagente coordenado. Eles interagem entre si para compartilhar informações, tomar decisões e fornecer respostas coesas ao usuário. Esta etapa envolve a sincronização das ações dos agentes e a gestão de seus estados e interações.
5. **Testes e Ajustes:** Após a implementação inicial, realizamos uma série de testes para avaliar o desempenho do chatbot. Isso inclui testar a precisão das respostas,

a eficiência da comunicação entre agentes e a usabilidade da interface. Com base nos resultados dos testes, fazemos os ajustes necessários para melhorar a funcionalidade e a experiência do usuário.

Através dessa abordagem de implementação, buscamos criar um chatbot que não apenas responda às perguntas dos usuários, mas também ofereça uma experiência interativa e enriquecedora, demonstrando os conceitos aprendidos na disciplina de Sistemas Especialistas.

## Fotos da Experiência

### 1. Codificação



The screenshot shows the chonIDE interface with a project named 'Multi-Agent System'. The 'Agents' folder is expanded, showing 'smith', 'papaleguas', and 'cliente'. The 'papaleguas' agent is selected, and its code is displayed in the editor. The code is written in a logic-oriented language and includes comments for initial beliefs, goals, and plans. It defines two main procedures: 'showmenu' and 'mostrarcardapio', which use print statements to display information and wait for user input. The 'showmenu' procedure lists menu options like 'Fazer um pedido', 'Ver cardápio', and 'Falar com um atendente'. The 'mostrarcardapio' procedure displays pizza sizes and toppings. The code is as follows:

```
1 /* Initial beliefs and rules */
2
3
4 /* Initial goals */
5
6 !showmenu.
7
8 /* Plans */
9
10 +!showmenu <-
11     .print("");
12     .print("");
13     .print("Ola, bem vindo ao atendimento aletronico da pizzeria Papaleguas");
14     .print("O que deseja ?");
15     .print("");
16     .print("1 - Fazer um pedido");
17     .print("2 - Ver cardápio");
18     .print("3 - Falar com um atendente");
19
20 +!mostrarcardapio <-
21     .print("");
22     .print("Temos os seguintes tamanhos: Familia, Media e pequena");
23     .print("Os sabores são os seguintes: ");
24     .print("- Calabresa");
25     .print("- 5 queijos");
26     .print("- Portuguesa");
27     .print("- Bacon");
28     .wait(5000);
29     .print("");
30     .print("Retornando para o menu inicial");
31     .wait(5000);
32     !showmenu.
33
34 +!chamaratendente <-
35     .print("");
36     .print("Para falar com um dos nossos atendentes entre em contato com o Controlador de volume 100")
```

Figura 1. Codificação 1

```
31 .wait(5000);
32 !showmenu.
33
34 +!chamaratendente <-
35     .print("");
36     .print("Para falar com um de nossos atendentes entre em contato com o número (21) 9 8541-8269");
37     .wait(5000);
38     .print("");
39     .print("Retornando para o menu inicial");
40     .wait(5000);
41     !showmenu.
42
43 +!fazerpedido <-
44     .print("");
45     .print("Escolha um dos tamanhos disponiveis");
46     .print("1 - Família");
47     .print("2 - Média");
48     .print("3 - Pequena");
49
50 +!escolhersabor <-
51     .print("");
52     .print("Agora escolha o sabor");
53     .print("- Calabresa");
54     .print("- 5 queijos");
55     .print("- Portuguesa");
56     .print("- Bacon");
57
58 +!encerrar: tamanho(Valor1) & sabor(Valor2) <-
59     .print("");
60     .print("O seu pedido foi uma pizza ", Valor1, " do sabor ", Valor2);
61     .print("Você será redirecionado para a página de pagamento.");
62     .print("");
63     .print("Retornando para o menu inicial");
64     .wait(3000);
65     !showmenu.
```

Figura 2. Codificação 2

```
1 /* Initial beliefs and rules */
2
3 /* Initial goals */
4
5 !start.
6
7 /* Plans */
8
9 +!start <-
10     .wait(10000);
11     .print("");
12     .print("2");
13     .print("");
14     .wait(2000);
15     .send(papaleguas, achieve, mostrarcardapio);
16     .wait(15000);
17     .print("");
18     .print("3");
19     .print("");
20     .wait(2000);
21     .send(papaleguas, achieve, chamaratendente);
22     .wait(15000);
23     .print("");
24     .print("1");
25     .print("");
26     .wait(2000);
27     .send(papaleguas, achieve, fazerpedido);
28     .wait(15000);
29     .print("");
30     .print("2");
31     .send(papaleguas, tell, tamanho("Media"));
32     .print("");
33     .wait(2000);
34     .send(papaleguas, achieve, escolhersabor);
35     .wait(15000);
36     .print("");
```

Figura 3. Codificação 3

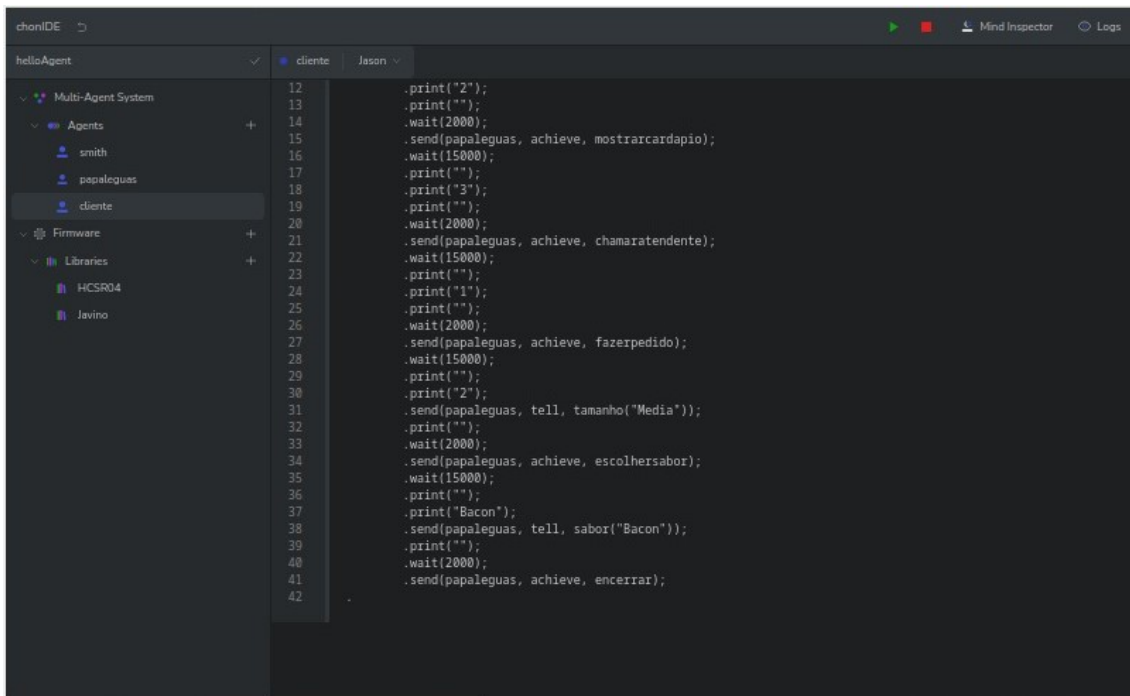


Figura 4. Codificação 4

## 2. Utilização

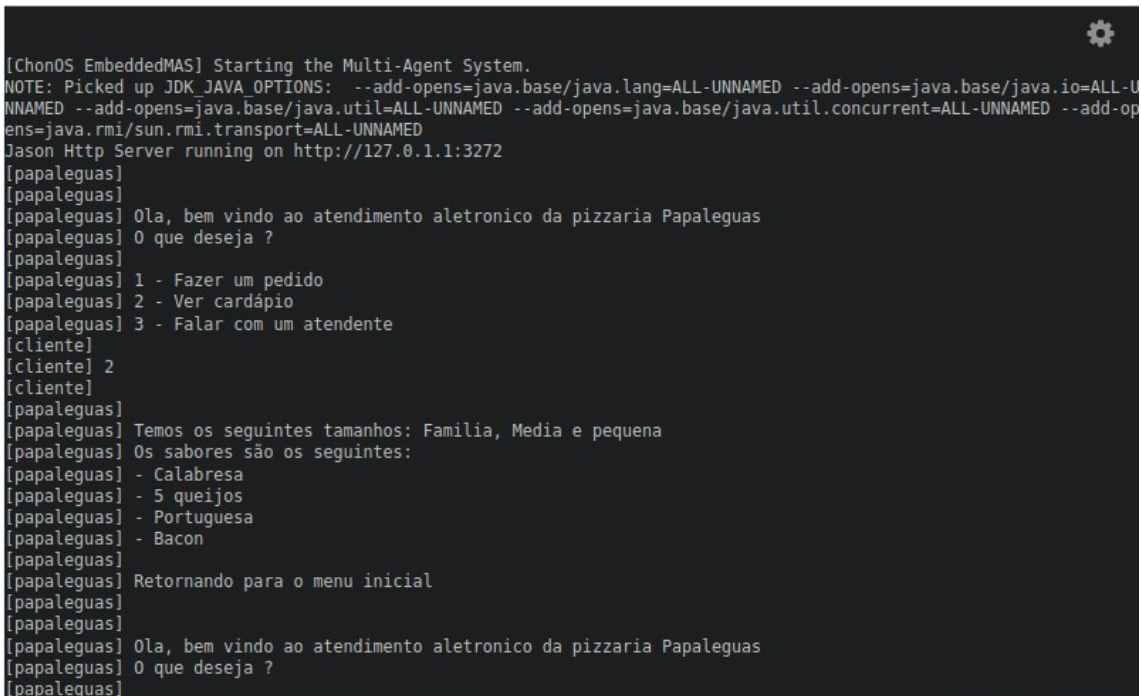


Figura 5. Utilização 1

```

[papaleguas]
[papaleguas]
[papaleguas] Ola, bem vindo ao atendimento aletronico da pizzeria Papaleguas
[papaleguas] 0 que deseja ?
[papaleguas]
[papaleguas] 1 - Fazer um pedido
[papaleguas] 2 - Ver cardápio
[papaleguas] 3 - Falar com um atendente
[cliente]
[cliente] 3
[cliente]
[papaleguas]
[papaleguas] Para falar com um de nossos atendentes entre em contato com o número (21) 9 8541-8269
[papaleguas]
[papaleguas] Retornando para o menu inicial
[papaleguas]
[papaleguas]
[papaleguas] Ola, bem vindo ao atendimento aletronico da pizzeria Papaleguas
[papaleguas] 0 que deseja ?
[papaleguas]
[papaleguas] 1 - Fazer um pedido
[papaleguas] 2 - Ver cardápio
[papaleguas] 3 - Falar com um atendente
[cliente]
[cliente] 1
[cliente]
[papaleguas]
[papaleguas] Escolha um dos tamanhos disponíveis
[papaleguas] 1 - Família
[papaleguas] 2 - Média
[papaleguas] 3 - Pequena
[cliente]
[cliente] 2

```

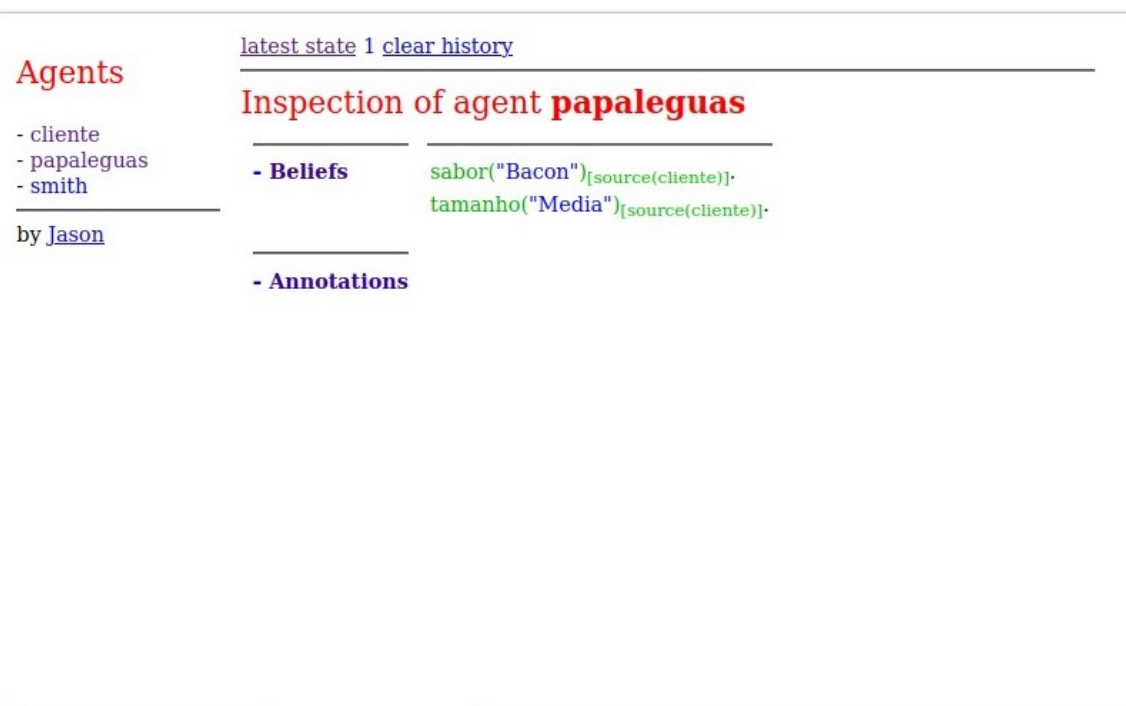
**Figura 6. Utilização 2**

```

[cliente]
[papaleguas]
[papaleguas] Escolha um dos tamanhos disponíveis
[papaleguas] 1 - Família
[papaleguas] 2 - Média
[papaleguas] 3 - Pequena
[cliente]
[cliente] 2
[cliente]
[papaleguas]
[papaleguas] Agora escolha o sabor
[papaleguas] - Calabresa
[papaleguas] - 5 queijos
[papaleguas] - Portuguesa
[papaleguas] - Bacon
[cliente]
[cliente] Bacon
[cliente]
[papaleguas]
[papaleguas] 0 seu pedido foi uma pizza Media do sabor Bacon
[papaleguas] Você será redirecionado para a página de pagamento.
[papaleguas]
[papaleguas] Retornando para o menu inicial
[papaleguas]
[papaleguas]
[papaleguas] Ola, bem vindo ao atendimento aletronico da pizzeria Papaleguas
[papaleguas] 0 que deseja ?
[papaleguas]
[papaleguas] 1 - Fazer um pedido
[papaleguas] 2 - Ver cardápio
[papaleguas] 3 - Falar com um atendente

```

**Figura 7. Utilização 3**



**Figura 8. Utilização 4**

## Referências

- [1] M. Wooldridge, *An Introduction to MultiAgent Systems*, John Wiley & Sons, 2009.
- [2] M. E. Bratman, *Intentions, Plans, and Practical Reason*, Harvard University Press, 1987.
- [3] Equipe Jason, *Documentação do Jason*, Disponível em: <http://jason.sourceforge.net/>, 2023, Acesso em: 27 de setembro de 2023.