

# Documentação Trabalho Prático

Bem-vindo à documentação técnica do Robô Social de Gerenciamento de Alarmes. Neste documento, você encontrará informações detalhadas sobre a arquitetura, as tecnologias utilizadas e o funcionamento do software.

## Arquitetura:

O Robô Social Gerenciamento de Alarmes foi desenvolvido utilizando a arquitetura Model-View-Controller (MVC), que permite uma organização eficiente do código e facilita a manutenção e a expansão do programa. Além disso, foram aplicados os conceitos aprendidos na disciplina de Inteligência Artificial referentes ao modelo Sense-Think-Act, que permitiu a criação de um robô social autônomo e natural.

No módulo de controle, podemos fazer uma analogia com o conceito de "Pensar" no modelo arquitetônico Sentir-Pensar-Agir. Aqui, encontram-se os arquivos que supervisionam a lógica que garante o funcionamento adequado do robô. O arquivo "control.py" é responsável por ativar as threads do programa relacionadas à interface e à interação do robô com alarmes. Isso permite que essas duas ações ocorram simultaneamente enquanto o programa está em execução. Além disso, os arquivos "definir\_alarme.py," "find\_horario.py," "loop\_alarme.py" e "variavel\_global.py" contêm funções cruciais para o desempenho do projeto, organizadas de forma sistemática dentro deste módulo.

No módulo de visualização, podemos estabelecer um paralelo com os conceitos de "Sentir" e "Agir" no modelo arquitetônico Sentir-Pensar-Agir. Este módulo abriga as funções responsáveis por receber informações e fornecer ao usuário os dados processados durante a operação do robô.

No arquivo "act\_disparar\_alarme.py," encontramos uma função que é acionada nos horários configurados para os alarmes. Os arquivos "stt.py" e "tts.py" implementam a integração com as APIs de conversão de discurso em texto e vice-versa. O arquivo "sense.py" contém a função que define o comportamento principal do robô durante a interação com o usuário. Por fim, em "interface.py," há uma função que cria uma interface gráfica exibida enquanto o robô está ativo, incluindo a exibição dos horários dos alarmes configurados pelo usuário.

## Tecnologias Utilizadas:

O software utiliza diversas tecnologias para garantir o seu funcionamento adequado. A seguir, listamos as principais:

- Python: linguagem de programação na versão 3.10 utilizada para desenvolver o software.

Pacotes/Bibliotecas/APIs:

- tkinter: pacote de interface gráfica de usuário para Python. Utilizada para criar a interface de visualização do projeto.
- schedule: API para tarefas de agendamento utilizada para o funcionamento da lógica do alarme.
- playsound: biblioteca de áudio utilizada para tocar o som no momento em que o alarme é acionado.
- requests: biblioteca utilizada para realizar requisições na API ElevenLabs Text-to-Speech.
- threading: pacote responsável por providenciar as funcionalidades de thread no programa.
- time: no projeto, essa biblioteca é importada para utilizar a função sleep() como forma de suspender a execução da thread por um breve período de tempo.
- datetime: o módulo datetime é utilizado para a manipulação de datas e horas em conjunto com a API schedule, definindo corretamente a configuração do alarme.
- re: biblioteca utilizada na lógica do arquivo find\_horario.py por meio da função findall() que retorna as correspondências não sobrepostas do padrão em string, com o objetivo de verificar horários na fala do usuário.
- PyAudio: biblioteca que provê funcionalidades de áudio em Python.
- Google Cloud Speech-to-Text API: API de reconhecimento de fala utilizada para permitir a interação por voz com o robô.
- ElevenLabs Text-to-Speech API: API de síntese de fala utilizada para permitir que o robô responda ao usuário por voz.

## Funcionamento:

O Robô Social de Gerenciamento de Alarmes possui as seguintes principais funcionalidades:

- O usuário pode visualizar a interface gráfica do programa.
- O robô reconhece a fala do usuário e processa as informações recebidas.
- O robô responde ao usuário por voz, confirmando a criação do alarme.
- O robô permite que o usuário agende um ou mais alarmes simultâneos.
- Quando um alarme é ativado, o robô emite um som e reproduz o lembrete gravado pelo usuário.

Em seguida, será descrito o fluxo de funcionamento do robô:

1. Robô é inicializado e é introduzido ao usuário como ativar o agendamento de alarme.
2. Robô aguarda a ativação do usuário através de voz pela palavra “Beto.”
3. O usuário, ao falar “Beto”, comunica ao robô que deseja configurar um alarme.
4. Robô reconhece o comando transmitido por voz do usuário.
5. Robô pergunta ao usuário qual o horário ele deseja agendar o alarme.
6. Usuário comunica ao robô o horário desejado.
7. Robô identifica o horário e o adiciona a uma lista de horários interna.
8. Robô solicita ao usuário que grave um lembrete para ser reproduzido junto da ativação do alarme.
9. Usuário comunica ao robô o que ele deseja gravar no lembrete.
10. Robô adiciona horário e lembrete como identificador do alarme na interface de visualização.
11. Após adicionar o alarme com sucesso, o robô volta a funcionar como no passo 2, sendo possível adicionar mais alarmes.
12. Ao atingir o horário de algum alarme adicionado, o robô reproduz áudio de alarme e lembrete gravado pelo usuário.