



Report

👤 Created by	👤 Silvio Gonçalves
🕒 Created time	@December 3, 2025 11:03 PM
☰ Category	Sprint 6
👤 Last edited by	👤 Silvio Gonçalves
🕒 Last updated time	@December 4, 2025 10:07 PM

The screenshot shows a dark-themed application interface for managing tasks. At the top, there is a header with the word "Tasks" and a checkmark icon. Below the header is a list of six tasks, each represented by a small document icon and bolded text. The tasks are:

- Desenvolver a codificação do ChatBot Front
- Desenvolver Telas Front (Home Page e Chat)
- Verificar o retorno do modelo, para evitar fallback
- Finalizar os endpoints back
- Integrar back com front
- Finalizar github Actions

At the bottom of the list, there are two buttons: "+ Add new" and "↗ Link existing".

- Nesta sprint, o principal desafio esteve na integração entre o frontend e o backend, especialmente na etapa de consumo dos endpoints pelo módulo de chat. Foram identificados conflitos de versionamento e incompatibilidades entre bibliotecas utilizadas em ambos os ambientes, impactando diretamente o fluxo de comunicação. Além disso, foi necessário orientar a equipe de frontend quanto à configuração obrigatória do arquivo `.env` contendo as credenciais da Google API, bem como o uso

do `credentials.json` específico para viabilizar a autenticação e a integração com o Google Calendar.

- O modelo apresentava um problema de baixa confiabilidade nas inferências: independentemente da consulta, a confiança retornada era insuficiente, acionando continuamente o fallback para o Gemini, mesmo quando as informações de medicação estavam devidamente armazenadas no ChromaDB. Após ajustes nos parâmetros de consulta e revisão do pipeline de recuperação, o comportamento inadequado foi sanado.
- Optamos por manter o projeto em dois repositórios distintos, mas necessitávamos garantir uma integração unificada para facilitar a avaliação pela banca. Para isso, configuramos um pipeline via GitHub Actions que realiza a sincronização automática (CI) do frontend para o repositório do backend. O processo exigiu um alinhamento inicial, já que os repositórios haviam sido configurados por pessoas diferentes, demandando a geração e validação de tokens no backend e sua posterior integração no frontend, além de ajustes finos no script de automação. Após essas configurações, o pipeline passou a operar corretamente, permitindo que qualquer atualização na branch `main` do frontend seja automaticamente refletida na branch `main` do backend.